



**You have downloaded a document from  
RE-BUS  
repository of the University of Silesia in Katowice**

**Title:** On the robust Newton's method with the Mann iteration and the artistic patterns from its dynamics

**Author:** Krzysztof Gdawiec, Wiesław Kotarski, Agnieszka Lisowska

**Citation style:** Gdawiec Krzysztof, Kotarski Wiesław, Lisowska Agnieszka. (2021). On the robust Newton's method with the Mann iteration and the artistic patterns from its dynamics. "Nonlinear Dynamics" Vol. 104 (2021), s. 297-331, doi 10.1007/s11071-021-06306-5



Uznanie autorstwa - Licencja ta pozwala na kopiowanie, zmienianie, rozprowadzanie, przedstawianie i wykonywanie utworu jedynie pod warunkiem oznaczenia autorstwa.



UNIwersYTET ŚLĄSKI  
W KATOWICACH



Biblioteka  
Uniwersytetu Śląskiego



Ministerstwo Nauki  
i Szkolnictwa Wyższego



# On the robust Newton's method with the Mann iteration and the artistic patterns from its dynamics

Krzysztof Gdawiec · Wiesław Kotarski ·  
Agnieszka Lisowska

Received: 28 August 2020 / Accepted: 11 February 2021 / Published online: 7 March 2021  
© The Author(s) 2021

**Abstract** There are two main aims of this paper. The first one is to show some improvement of the robust Newton's method (RNM) introduced recently by Kalantari. The RNM is a generalisation of the well-known Newton's root finding method. Since the base method is undefined at critical points, the RNM allows working also at such points. In this paper, we improve the RNM method by applying the Mann iteration instead of the standard Picard iteration. This leads to an essential decrease in the number of root finding steps without visible destroying the sharp boundaries among the basins of attractions presented in polynomiographs. Furthermore, we investigate visually the dynamics of the RNM with the Mann iteration together with the basins of attraction for varying Mann's iteration parameter with the help of polynomiographs for several polynomials. The second aim of this paper is to present the intriguing polynomiographs obtained from the dynamics of the RNM with the Mann iteration under various sequences used in this iteration. The obtained polynomiographs differ considerably from the ones obtained with the RNM and are interesting from

the artistic perspective. Moreover, they can easily find applications in wallpaper or fabric design.

**Keywords** Robust Newton method · Mann iteration · Polynomiography · Dynamics · Artistic patterns

## 1 Introduction

The Newton's method is one of the most commonly known numerical algorithms—that at first was proposed around 1669 by Newton and then modified in 1690 by Raphson—for finding real roots of polynomials. Further, Simpson (1740) extended the method to solve nonlinear equations, and Cayley (1879) used the method for finding complex roots of polynomials. Cayley faced a strange and unpredictable chaotic behaviour—that he could not explain—of Newton's method in the case of a simple equation  $z^3 - 1 = 0$  in the complex plane. The solution to that problem was then found in 1919 by Julia. Julia sets led next Mandelbrot in the 1970s to the discovery of the famous Mandelbrot set and fractals. Newton's method is also applied in optimisation for finding a local extreme of a differentiable function that is a root of its gradient [28]. The Newton's method, depending on a starting point, might be convergent or divergent, has a local quadratic convergence and is undefined for critical points (i.e. points for which the first derivative of a function equals to 0). This method is sensitive to a starting point and

---

K. Gdawiec (✉) · W. Kotarski · A. Lisowska  
Institute of Computer Science, University of Silesia,  
Będzińska 39, 41-200 Sosnowiec, Poland  
e-mail: krzysztof.gdawiec@us.edu.pl

W. Kotarski  
e-mail: kotarski@ux2.math.us.edu.pl

A. Lisowska  
e-mail: agnieszka.lisowska@us.edu.pl

finds stable or unstable cyclic values. So then, Newton's method dynamic behaviour is very rich [16,34].

In the literature [15,23,24,30,35] there are known many modifications and improvements of Newton's method. Here, we would like to point out some of the most recent results in this area. In [10] various modifications of Newton's method with arithmetic, harmonic, and contraharmonic means are investigated. In [3,8,14] Newton's method was modified by using fractional derivatives instead of the classical ones. Wang and Tao [33] proposed a new way to construct a self-accelerating parameter in Newton's method with memory.

The most striking effects that can be observed when finding the roots of the polynomial  $z^3 - 1$  on the complex plane are the fractal boundaries among the basins of attractions caused by chaos and instability. In the case of complex polynomials, the Newton method, producing Newton fractals, is an important model of deterministic chaos [35] similarly as the logistic equation in the real case [22]. Basins of attraction form stunning images that are ones of the most beautiful fractals. Fractal boundaries in the form of braids can be decreased or even, to some extent, eliminated using the damped Newton method [11]. Recently, Kalantari [19] proposed the Robust Newton Method (RNM) that radically smooths the boundaries among basins of attraction that form sharp lines. That result is obtained by precise controlling of the step's length in the Newton method. Unfortunately, the RNM usually needs a very large number of iterations. Kalantari [19] showed that for a given accuracy  $\varepsilon > 0$  the number of iterations needed by the RNM to converge to the root or critical point is  $\mathcal{O}(1/\varepsilon^2)$ . By using this bound, for example, for  $\varepsilon = 0.01$ , we obtain  $1/\varepsilon^2 = 10,000$ . Thus, even for not so accurate computations for some points, we need to perform a large number of iterations, and this number increases with the increase in the accuracy. This can be treated as a drawback of the RNM.

In this paper, we propose the modification of the RNM, which relies on replacing the Picard iteration used in the RNM by the Mann iteration. This modification will decrease the average number of iterations needed to find the solution and, at the same time, the method will preserve its other features. We will analyse numerically different aspects of the proposed modification, e.g. basins of attraction, dynamics, and the average number of iterations. Moreover, by using various sequences of parameters in the Mann iteration, we will

generate very complex and intriguing patterns from the modified method dynamics. These patterns might have artistic applications.

The paper is organised as follows. Section 2 presents the description of the RNM. In Sect. 3, modification of the RNM with the Mann iteration is given. Then, in Sect. 4, the plots of average number of iterations (ANI), convergence area index (CAI) and time generation characterising the RNM with Mann iteration are given. Also, polynomiographs presenting basins of attraction and dynamic properties of the considered root finding processes for a given set of polynomials are shown. Section 5 is devoted to artistic aspects of the RNM with Mann iteration related to the generation of very complex and beautiful polynomiographs from dynamics. The last section, Sect. 6, concludes the paper and points out future directions of research.

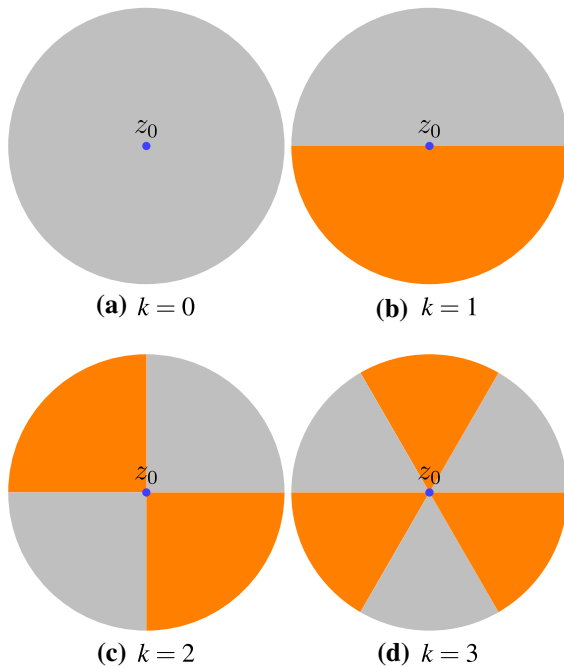
## 2 The robust Newton's method

The robust Newton method (RNM) was proposed to overcome the Newton's method main drawback—the lack of definition at critical points [19]. The RNM guarantees the reduction of the polynomial's modulus in successive iterations and has several differences compared to the classical Newton's method. Firstly, it converges globally, whereas the classical Newton's method converges locally. Secondly, it finds roots and critical points of polynomials, whereas classical Newton's method finds only the roots.

The RNM is based on the following facts:

- Instead of finding zeros for polynomial  $p(z)$ , the equivalent modulus minimisation of  $F(z) = |p(z)|^2 = p(z)\overline{p(z)}$  is considered. Observe that minimisation of  $|p(z)|$  and  $F(z)$  are also equivalent.
- In the classical Newton's method applied to  $F(z)$  at every iteration, a descent direction is chosen according to the geometric modulus principle (GMP) [18], and the step size is taken on the base of the modulus reduction theorem (MRT) [19].

The GMP gives a complete characterisation of all descent and ascent directions for  $|p(z_0)|$  at an arbitrary point  $z_0$  and additionally reveals a surprising geometric pattern showing partition of a unit disc, centred at  $z_0$ , into sectors of ascent and descent directions. The GMP for polynomials [18] says that if a non-constant polynomial  $p(z)$  at  $z_0$  equals to zero, then every direction



**Fig. 1** Sectors of ascent (grey colour) and descent (orange colour) directions for various values of  $k$ . (Color figure online)

at  $z_0$  is an ascent direction for  $|p(z_0)|$ . If  $p(z_0) \neq 0$  then the cones of ascent and descent directions at  $z_0$  divide the unit disc centred at  $z_0$  into alternating ascent and descent sectors of equal angle  $\pi/k$ , where  $k \geq 1$  is the smallest index with  $p^{(k)}(z_0) \neq 0$ . Such typical disc partitionings, except for possible rotations, for  $k = 0, 1, 2, 3$  are presented in Fig. 1.

Next, the MRT assures a reduction in the modulus of  $p(z)$  when moving from  $z_0$  to  $z_1$  by the amount proportional to  $|p(z_0)p'(z_0)|^2$ . Summing up, the GMP enables to choose consequently a descent direction, whereas the MRT guarantees the decrease of  $F(z)$  along the chosen descent direction at every iteration step of the RNM. Here, it is worth to add that a random direction at  $z_0$  has a fifty per cent chance to be a descent (or ascent) direction—what stresses the great importance of the GMP.

In the sequel, we recall the formulas defining the RNM, following [19].

Let us consider a complex polynomial  $p$ :

$$p(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0, \quad (1)$$

where  $z \in \mathbb{C}$ ,  $a_n, a_{n-1}, \dots, a_0 \in \mathbb{C}$ , and  $n \in \mathbb{N}$ .

Assume that  $p(z) \neq 0$ ,  $z \in \mathbb{C}$  and let us define [19]:

$$k \equiv k(z) = \min\{j \geq 1 : p^{(j)}(z) \neq 0\}, \quad (2)$$

$$A(z) = \max \left\{ \frac{|p^{(j)}(z)|}{j!} : j = 0, \dots, n \right\}, \quad (3)$$

$$u_k \equiv u_k(z) = \frac{1}{k!} p(z) \overline{p^{(k)}(z)}. \quad (4)$$

Moreover, let us define:

$$\gamma_k = 2 \cdot \operatorname{Re} \left( u_k^{k-1} \right), \quad (5)$$

$$\delta_k = -2 \cdot \operatorname{Im} \left( u_k^{k-1} \right), \quad (6)$$

$$c_k = \max\{|\gamma_k|, |\delta_k|\}. \quad (7)$$

Additionally, let  $\theta_k$  be the angle given by the following formula:

$$\theta_k = \begin{cases} 0, & \text{if } c_k = |\gamma_k| \wedge \gamma_k < 0, \\ \pi/k, & \text{if } c_k = |\gamma_k| \wedge \gamma_k > 0, \\ \pi/(2k), & \text{if } c_k = |\delta_k| \wedge \delta_k < 0, \\ 3\pi/(2k), & \text{if } c_k = |\delta_k| \wedge \delta_k > 0. \end{cases} \quad (8)$$

Now, the RNM for the starting point  $z_0 \in \mathbb{C}$  is defined as:

$$z_{i+1} = N_p(z_i), \quad i = 0, 1, 2, \dots, \quad (9)$$

where

$$N_p(z_i) = z_i + \frac{C_k(z_i)}{3} \frac{u_k}{|u_k|} e^{i\theta_k} \quad (10)$$

and  $i$  denotes the imaginary unit, i.e.  $i = \sqrt{-1}$ , and

$$C_k(z_i) = \frac{c_k |u_k|^{2-k}}{6A^2(z_i)}. \quad (11)$$

The term  $(u_k/|u_k|)e^{i\theta_k}$  is called as the normalised robust Newton direction at  $z_i$  [19] and  $C_k(z_i)/3$  is the step size.

Because the RNM finds roots and critical points, the stopping criterion has some other form than the one for the classical Newton's method, and is given by the following condition:

$$|p(z_i)| < \varepsilon \vee |p'(z_i)| < \varepsilon, \quad (12)$$

where  $\varepsilon > 0$  is the accuracy.

To summarise the RNM, in Algorithm 1, we present its pseudocode, and in Algorithm 2, the pseudocode for the  $N_p(z)$  computation.



**Algorithm 1:** The Robust Newton Method.

**Input:**  $p$  – complex polynomial;  $z_0 \in \mathbb{C}$  – the starting point;  $M$  – the maximum number of iterations;  $\varepsilon > 0$  – accuracy.

**Output:** A root or critical point of  $p$ . If the RNM does not find the solution in  $M$  iterations, the algorithm throws an exception.

```

1  $i = 0$ 
2 while  $i < M$  do
3   if  $|p(z_i)| < \varepsilon \vee |p'(z_i)| < \varepsilon$  then
4     return  $z_i$ 
5    $z_{i+1} = N_p(z_i)$ 
6    $i = i + 1$ 
7 Throw “No solution found.”

```

**3 The robust Newton’s method with the Mann iteration**

The problem of finding a fixed point of a given mapping is well-studied in the literature [1, 6]. In these studies, one of the directions is using iterative approximation methods for finding the fixed points. The most famous method is the Picard iteration [27]:

$$z_{i+1} = T(z_i), \quad i = 0, 1, 2, \dots, \quad (13)$$

where  $T$  is a given mapping for which we search fixed points and  $z_0 \in \mathbb{C}$  is a starting point.

Another known iteration is the Mann iteration (defined in 1953 by William Robert Mann) [21]:

$$z_{i+1} = (1 - \alpha_i)z_i + \alpha_i T(z_i), \quad i = 0, 1, 2, \dots, \quad (14)$$

where  $\alpha_i \in (0, 1]$ . Depending on the type of operator  $T$  (contractive, expanding, etc.) and the space (Banach, hyperbolic, CAT(0), etc.) the sequence  $\alpha_i$  has to fulfil some additional conditions in order (14) to be convergent to a fixed point.

Let us notice that the Mann iteration for  $\alpha_i = 1$  for all  $i$  reduces to the Picard iteration. In the literature, one can find many other iterations. A review of 18 different iterations and their dependencies can be found in [13].

At this point, we combine the RNM with the Mann iteration. We do so by using  $N_p$  as  $T$  in (14) and obtaining:

$$z_{i+1} = (1 - \alpha_i)z_i + \alpha_i N_p(z_i), \quad i = 0, 1, 2, \dots \quad (15)$$

From now on, we will call the RNM with the Mann iteration, shortly, as M-RNM.

**Algorithm 2:** Computation of  $N_p$ .

**Input:**  $p$  – complex polynomial of degree  $n$ ;  $z \in \mathbb{C}$ ;  $\varepsilon > 0$  – accuracy.

**Output:**  $N_p(z)$ .

```

1  $k = 1$ 
2 while  $|p^{(k)}(z)| < \varepsilon \wedge k + 1 \leq n$  do
3    $k = k + 1$ 
4    $u_k = \frac{1}{k!} p(z) \overline{p^{(k)}(z)}$ 
5    $A = \max \left\{ |p(z)|, \frac{|p^{(k)}(z)|}{k!} \right\}$ 
6    $j = k$ 
7   while  $j + 1 \leq n$  do
8      $j = j + 1$ 
9      $A = \max \left\{ A, \frac{|p^{(j)}(z)|}{j!} \right\}$ 
10   $\gamma_k = 2 \cdot \operatorname{Re}(u_k^{k-1})$ 
11   $\delta_k = -2 \cdot \operatorname{Im}(u_k^{k-1})$ 
12  if  $|\gamma_k| \geq |\delta_k|$  then
13     $c_k = |\gamma_k|$ 
14    if  $\gamma_k < 0$  then
15       $\theta_k = 0$ 
16    else
17       $\theta_k = \frac{\pi}{k}$ 
18  else
19     $c_k = |\delta_k|$ 
20    if  $\delta_k < 0$  then
21       $\theta_k = \frac{\pi}{2k}$ 
22    else
23       $\theta_k = \frac{3\pi}{2k}$ 
24   $C_k = \frac{c_k |u_k|^{2-k}}{6A^2}$ 
25 return  $z + \frac{C_k u_k}{3|u_k|} e^{i\theta_k}$ 

```

Now, let us write (15) in the following form:

$$z_{i+1} = (1 - \alpha_i)z_i + \alpha_i N_p(z_i) = z_i + \alpha_i (N_p(z_i) - z_i). \quad (16)$$

By looking at this formula, we can observe that at the  $(i + 1)$ -th iteration the M-RNM moves  $z_i$  into the direction given by vector  $N_p(z_i) - z_i$  scaled by  $\alpha_i$ . Let us note that  $N_p(z_i)$  moves  $z_i$  towards the solution assuring reduction in the modulus of  $p$ , so  $N_p(z_i) - z_i$  points in the same direction. Now, everything depends on  $\alpha_i$ . If  $\alpha_i < 0$ , then the point moves in the direction opposite to the modulus reduction. If  $\alpha_i > 0$ , the point moves towards the modulus reduction, but, depending on the value of  $\alpha_i$ , this point can move faster or slower, i.e. the modulus reduction is larger or smaller or if the value  $\alpha_i$  is too high one can obtain an increase in the modulus.

**Table 1** Lengths of the orbits presented in Fig. 2

$\alpha$	Orbit's length
0.2	408
1.0	80
2.0	39
5.0	15
9.32	6
15.0	20
19.65	501

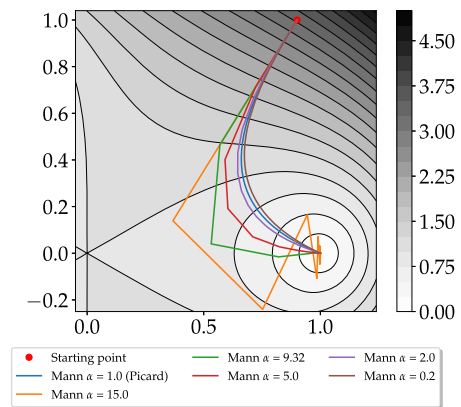
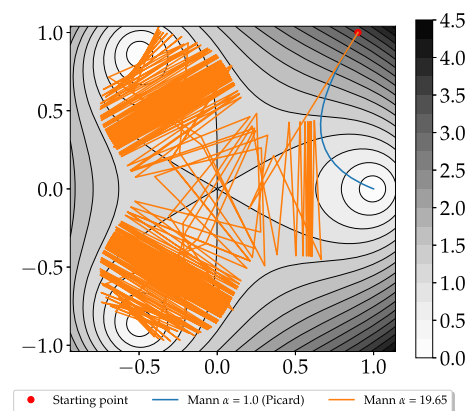
After including (10) into (16), we obtain the following:

$$\begin{aligned}
 z_{i+1} &= z_i + \alpha_i (N_p(z_i) - z_i) \\
 &= z_i + \alpha_i \left( z_i + \frac{C_k(z_i)}{3} \frac{u_k}{|u_k|} e^{i\theta_k} - z_i \right) \quad (17) \\
 &= z_i + \alpha_i \frac{C_k(z_i)}{3} \frac{u_k}{|u_k|} e^{i\theta_k}.
 \end{aligned}$$

From this form of the M-RNM, it is easily seen that we obtained the RNM with an additional damping factor  $\alpha_i$ . If we join  $\alpha_i$  with the term representing the step size, i.e.  $C_k(z_i)/3$ , we see that  $\alpha_i$  changes the step size in the original RNM.

To see how  $\alpha_i$  affects the root finding process of the RNM let us consider the following example. We search for the roots of  $p(z) = z^3 - 1$  and we use  $\alpha_i = \text{const} = \alpha$ . We presented the orbits for the starting point  $z_0 = 0.9 + i$  in Fig. 2 for various values of  $\alpha$ . We fixed the maximum number of iterations to 500. According to our previous observation, the orbit for  $\alpha = 1$  presents the orbit for the RNM introduced in [19]. When we look at the various orbits, we can see that for  $\alpha = 1$ , the orbit forms a smooth curve. If we increase the value of  $\alpha$ , we see that the curve becomes a polyline. In the plot presented in Fig. 2a, we can see that for  $\alpha > 1.0$ , we obtain a greater modulus reduction and the length of the corresponding orbit is shorter (see Table 1) than in the case of  $\alpha = 1$ . However, if the value of  $\alpha$  is too high, then the orbit's points jump from the neighbourhood of one root to the neighbourhood of the other root and, as a consequence, this method does not find the solution (see Fig. 2b). Therefore, from this example, we see that by a proper selection of  $\alpha$  we can make (15) faster than the original method proposed in [19].

The selection of  $\alpha$  is not an easy task, because the best value of  $\alpha$  is different for different polynomials and

**(a)****(b)**

**Fig. 2** Contour plots of  $|p(z)| = |z^3 - 1|$  and the orbits for the M-RNM for various values of  $\alpha$

there is no obvious dependency between  $\alpha$ , polynomial  $p$  and the orbit's length. In each step, we could try to find the best value of  $\alpha$  by minimising the modulus, i.e.

$$\begin{aligned}
 \alpha_i &= \arg \min_{\alpha > 0} |p((1 - \alpha)z_i + \alpha N_p(z_i))| \\
 &= \arg \min_{\alpha > 0} |p(z_{i+1})|. \quad (18)
 \end{aligned}$$

This is a similar equation as in the gradient descent method with the exact line search [31]. By using (18), we can always find  $\alpha$  that assures modulus reduction. This is since for  $\alpha = 1$ , we get the RNM which reduces the modulus [19]. Therefore, if, in the current step, we do not find  $\alpha \neq 1$  that reduces the modulus better than the RNM, then we use RNM ( $\alpha = 1$ ). This method may give the best value of  $\alpha$ , but it is computationally too expensive in practice. Instead, we can use a method similar to the backtracking line search known

in optimisation [31]. The proposed method is presented in Algorithm 3.

---

**Algorithm 3:** Finding of the best value of  $\alpha$  in the Mann iteration for  $(i + 1)$ -th step.

---

**Input:**  $p$  – complex polynomial;  $z_i$  – root approximation from the  $i$ -th step;  $\alpha$  – the starting value;  
 $s \in (0, 1)$  – the scaling factor.

**Output:** The value of  $\alpha$  for  $(i + 1)$ -th step.

```

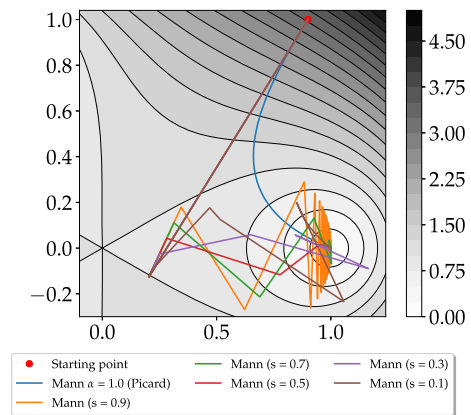
1  $\alpha_i = \alpha$ 
2 while  $|p((1 - \alpha_i)z_i + \alpha_i N_p(z_i))| > |p(z_i)|$  do
3    $\alpha_i = s\alpha_i$ 
4 return  $\alpha_i$ 

```

---

Using the same example as in the case of the constant  $\alpha$ , i.e. the contour plot of  $p(z) = z^3 - 1$ , let us see how the orbits look like when we use Algorithm 3. In this example, we take 19.65 as the starting value of  $\alpha$ , and we generate orbits for various values of  $s$ . The obtained orbits are presented in Fig. 3. The starting value of  $\alpha$  is the same as for the constant  $\alpha$  used in Fig. 2b, where the orbit has jumped between neighbourhoods of the roots. After using Algorithm 3, we see that the orbits do not jump. Instead, they tend towards the root. For various values of  $s$ , we can see that the first step is the same, but the orbits differ from the second step. The lengths of the orbits are gathered in Table 2. From the data, we can see that the orbits for  $s$  equal to 0.1, 0.3, 0.5 and 0.7 are much shorter than in the case of the RNM (orbit's length equal to 80). Only for  $s = 0.9$  we obtain longer orbit, but the method has converged in contrast to the constant  $\alpha$  value equal to 19.65.

By introducing the Mann iteration in the RNM, we can ask whether the new method is convergent or not. The answer is affirmative when we assume that  $\alpha_i$  is selected so that in each iteration we get modulus reduction. We can make such an assumption, because if in



**Fig. 3** Contour plots of  $|p(z)| = |z^3 - 1|$  and the orbits for the M-RNM for various values of  $s$ , and the starting value of  $\alpha = 19.65$  according to Algorithm 3

the current iteration we do not find  $\alpha \neq 1$  that reduces the modulus, then we can use  $\alpha = 1$ , i.e. the standard RNM for which it has been proven that it reduces the modulus [19]. In this situation, by using the M-RNM for any starting point, we obtain a decreasing sequence bounded below, so it is convergent.

## 4 Numerical results

In this section, we present the numerical results obtained with the help of the method proposed in Sect. 3 for  $\alpha_i = \text{const} = \alpha$  and for  $\alpha_i$  computed by using Algorithm 3. The results will be divided into two categories. The first category consists of images showing basins of attraction and dynamics, the so-called polynomiography [17]. In the second category, we will present various numerical measures, such as the average number of iterations [4], convergence area index [5] and polynomiograph's generation time [12].

Nowadays, polynomiography is an essential part of modern analysis of the quality of the root-finding methods [26]. Depending on the selected colouring method, we can visualise various aspects of the considered root finding method with the help of polynomiography. There are two standard methods of colouring polynomiographs.

In the first method, called basins of attraction, each root has its distinct colour, but other than some fixed colour, usually black, that represents the case in which the method has not converged to any root. Then, for the last point  $z_{i+1}$ , we search for the closest root (by

**Table 2** Lengths of the orbits presented in Fig. 3

$s$	Orbit's length
0.1	35
0.3	13
0.5	7
0.7	15
0.9	181

using the modulus metric) and colour the starting point with the colour of the root if the number of performed iterations was less than  $M$ , and with the fixed colour (black) otherwise. For the RNM and M-RNM, we add one more colour to colour the critical points. The algorithm for generating basins of attraction is presented in Algorithm 4.

---

**Algorithm 4:** Polynomiograph generation – basins of attraction.

---

**Input:**  $p$  – complex polynomial;  $\{r_1, r_2, \dots, r_n\}$  – the roots of  $p$ ;  $A \subset \mathbb{C}$  – area;  $M$  – the maximum number of iterations;  $\varepsilon > 0$  – accuracy;  $\alpha_i \in \mathbb{R}_+$  – the parameters sequence in the Mann iteration;  $colours[1..n]$  – colours for the roots;  $c_{cr}$  – the colour for critical points;  $c_{nc}$  – the colour for non-convergent points.

**Output:** Polynomiograph for the area  $A$ .

```

1 for  $z_0 \in A$  do
2    $i = 0$ 
3   while  $i < M$  do
4     if  $|p(z_i)| < \varepsilon \vee |p'(z_i)| < \varepsilon$  then
5       break
6      $z_{i+1} = (1 - \alpha_i)z_i + \alpha_i N_p(z_i)$ 
7      $i = i + 1$ 
8   if  $i < M$  then
9      $isRoot = false$ 
10    for  $j = 1, \dots, n$  do
11      if  $|z_i - r_j| < \varepsilon$  then
12        colour  $z_0$  with  $colours[j]$ 
13         $isRoot = true$ 
14        break
15    if  $isRoot = false$  then
16      colour  $z_0$  with  $c_{cr}$ 
17  else
18    colour  $z_0$  with  $c_{nc}$ 

```

---

In the second method, the so-called iteration colouring, which depicts the speed of convergence and the dynamics of the considered method, the starting point is coloured by linearly mapping the number of performed iterations on the colour in the given colour map. The algorithm for generating this type of polynomiographs is presented in Algorithm 5.

As we mentioned at the beginning of this section, we used three numerical measures in our comparisons. The first measure is the average number of iterations (ANI) [4]. To compute ANI, we need a polynomiograph generated with the iteration colouring. In this polynomio-

---

**Algorithm 5:** Polynomiograph generation – speed of convergence and dynamics.

---

**Input:**  $p$  – complex polynomial;  $A \subset \mathbb{C}$  – area;  $M$  – the maximum number of iterations;  $\varepsilon > 0$  – accuracy;  $\alpha_i \in \mathbb{R}_+$  – the parameters sequence in the Mann iteration;  $colours[0..N]$  – colour map.

**Output:** Polynomiograph for the area  $A$ .

```

1 for  $z_0 \in A$  do
2    $i = 0$ 
3   while  $i < M$  do
4     if  $|p(z_i)| < \varepsilon \vee |p'(z_i)| < \varepsilon$  then
5       break
6      $z_{i+1} = (1 - \alpha_i)z_i + \alpha_i N_p(z_i)$ 
7      $i = i + 1$ 
8    $j = \lfloor N \frac{i}{M} \rfloor$ 
9   colour  $z_0$  with  $colours[j]$ 

```

---

graph, each point corresponds to the number of iterations needed to find a root. As the name suggests, ANI is computed as the average number of iterations in the given polynomiograph. The second measure we use is the convergence area index (CAI) [5]. This measure can be computed from basins of attraction or a polynomiograph generated with the iteration colouring. CAI is the ratio of the number of convergent points  $N_c$  to the number of all points  $N$  in the considered polynomiograph, i.e.

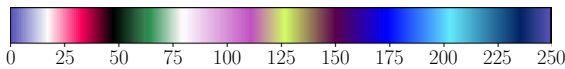
$$CAI = \frac{N_c}{N}. \quad (19)$$

From the definition of CAI we can see that it is between 0 and 1, and it gives us information about the percentage of polynomiograph's area that has converged to roots. The last considered measure is the time of generation of the polynomiograph [12]. It gives us information about the real time of computations.

In the numerical examples presented in this section, we used the following three polynomials, which are the most commonly used in the literature:

- $p(z) = z^3 - 1$  with the roots:  $-\frac{1}{2} - \frac{\sqrt{3}}{2}i, -\frac{1}{2} + \frac{\sqrt{3}}{2}i, 1$ ,
- $p(z) = z^4 - 1$  with the roots:  $-1, 1, -i, i$ ,
- $p(z) = z^5 + z$  with the roots:  $0, -\frac{1+i}{\sqrt{2}}, \frac{1+i}{\sqrt{2}}, -\frac{1-i}{\sqrt{2}}, \frac{1-i}{\sqrt{2}}$ .

To generate the polynomiographs, we used the following parameters:  $A = [-3, 3]^2$ ,  $M = 250$ ,  $\varepsilon = 0.001$ , and image resolution of  $800 \times 800$  pixels. The colour map used for the iteration colouring is presented



**Fig. 4** Colour map used in the dynamics polynomiographs with denoted numbers of steps

in Fig. 4. In the basins of attraction, the lack of convergence is depicted by the black colour, whereas the critical points by yellow colour and the roots by any other ones. In the experiments with the constant  $\alpha$  parameter used in the Mann iteration, the  $\alpha$  was taken from  $(0, 50]$  with the step equal to 0.01. In case of the variable  $\alpha$  parameter, i.e. computed with Algorithm 3, several values for the starting  $\alpha$  were investigated and the  $s$  parameter was taken from  $(0, 1)$  with the step equal to 0.01.

All experiments were performed on the computer with the following specifications: Intel i5-9600K (@ 3.70 GHz), 32 GB DDR4 RAM, NVIDIA GeForce GTX 1660 Ti graphics card with 6 GB GDDR6 SDRAM and Windows 10 (64 bit). The software for polynomiograph's generation was implemented in the C++ programming language with OpenGL (Open Graphics Library) and GLSL (OpenGL Shading Language). The computations of polynomiographs were performed on the graphics card with the use of shaders written in GLSL.

$$4.1 \quad p(z) = z^3 - 1$$

#### 4.1.1 Constant $\alpha$ sequence

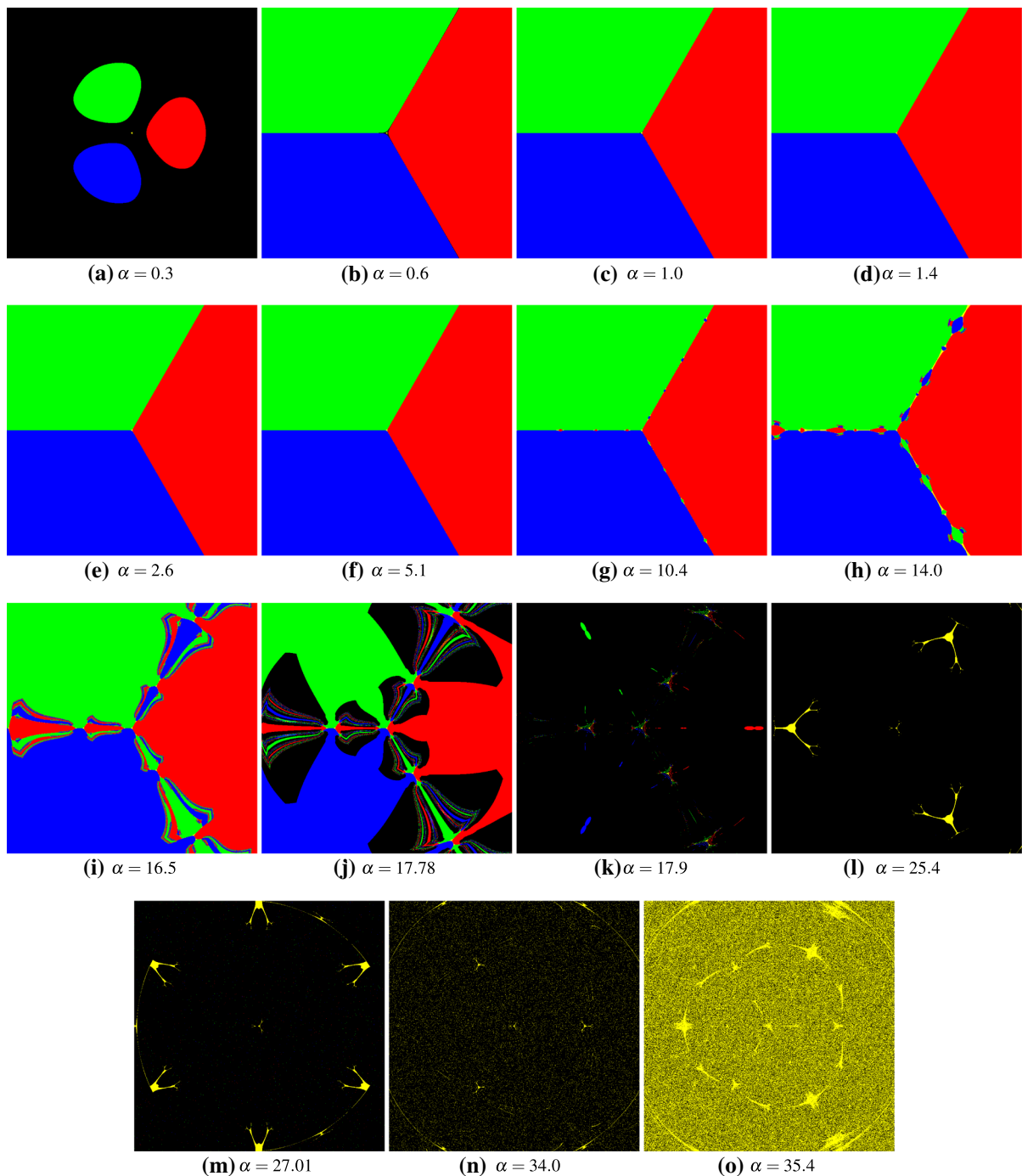
In Fig. 5, the basins of attraction for  $p(z) = z^3 - 1$  for different values of  $\alpha$  are presented. Since this polynomial has three roots and one critical point, we can see four corresponding basins. Let us note that for  $\alpha = 1.0$ , we cope with the Picard's iteration. Thus, we obtained the polynomiograph for the RNM. The boundaries of the basins are sharp and not fractal-like in the case of classical Newton's method. Moreover, in the centre of this polynomiograph (point  $0 + 0i$ ), there is a yellow dot. This is the critical point of  $p$ . Furthermore, we can see that the decrease in  $\alpha$  causes that fewer points converge to the roots. Indeed, when  $\alpha$  tends to zero, the black colour floods from the proximity of the critical point until the total vanishing of the red, green, and blue areas. The yellow dot in the centre stays forever. On the other hand, the increase in  $\alpha$  causes

something different and the dynamics of changes is also different. We can observe that starting from about  $\alpha = 10.4$ , the basins' regularity is broken at the borders and tends to be larger. From  $\alpha = 17.78$  the number of convergent points dramatically changes and, finally, these points disappear. Instead, some points converge to the critical point (yellow areas). From about  $\alpha = 34.0$  the points that converge to the critical point start to appear randomly.

In Fig. 6, the dynamics for the same polynomial and the same values of  $\alpha$  are presented. In such a way, we can observe how fast the proposed method converges for the plotted starting points according to the colour map from Fig. 4. However, because we have chosen the colour map that is not monotonic, and we did it to emphasize the dynamics and clearly see the differences, it is a little bit difficult to see the increase in the speed of convergence for appropriate points without looking at the colour map. However, yet one can observe that, as in the case of basins of attraction, for the same values of  $\alpha$  the same points are non-convergent (black areas). Additionally, for  $\alpha = 1.0$  one, can see that the closer the point is to one of the roots, the faster the convergence is (what is in accordance with intuition). When we tend with  $\alpha$  to zero, we see that the colours tend to move to the right on the colour map (see Fig. 4). It means that the overall convergence becomes slower. On the other hand, when we increase  $\alpha$  we move to the left on the colour map, so the overall convergence becomes faster. However, from about  $\alpha = 10.4$ , the convergence becomes slower again. This is the visual confirmation of the facts discussed in Sect. 3 that depending on the value of  $\alpha$  the speed of convergence changes, and we see here that this is done globally for all points.

To make the considerations complete, in Fig. 7, the plots of the commonly used measures: ANI, CAI and generation time (in seconds) are presented. From Fig. 7a, one can see that the average number of iterations is the smallest for  $\alpha = 9.32$  and equals to 5.63, which is considerably smaller than in the case of the RNM for which the value of ANI is equal to 84.89. For  $\alpha \in [17.9, 33.74]$ , we get nearly maximal number of iterations. Then, for  $\alpha > 33.74$ , the average number of iterations changes randomly and has a quite large magnitude. By observing index CAI (representing the normalised convergence area), one can see that for the RNM method ( $\alpha = 1.0$ ), we have a high CAI value, i.e. about 0.99, which means that some points have not converged. From  $\alpha = 1.74$  up to  $\alpha = 17.75$ , all

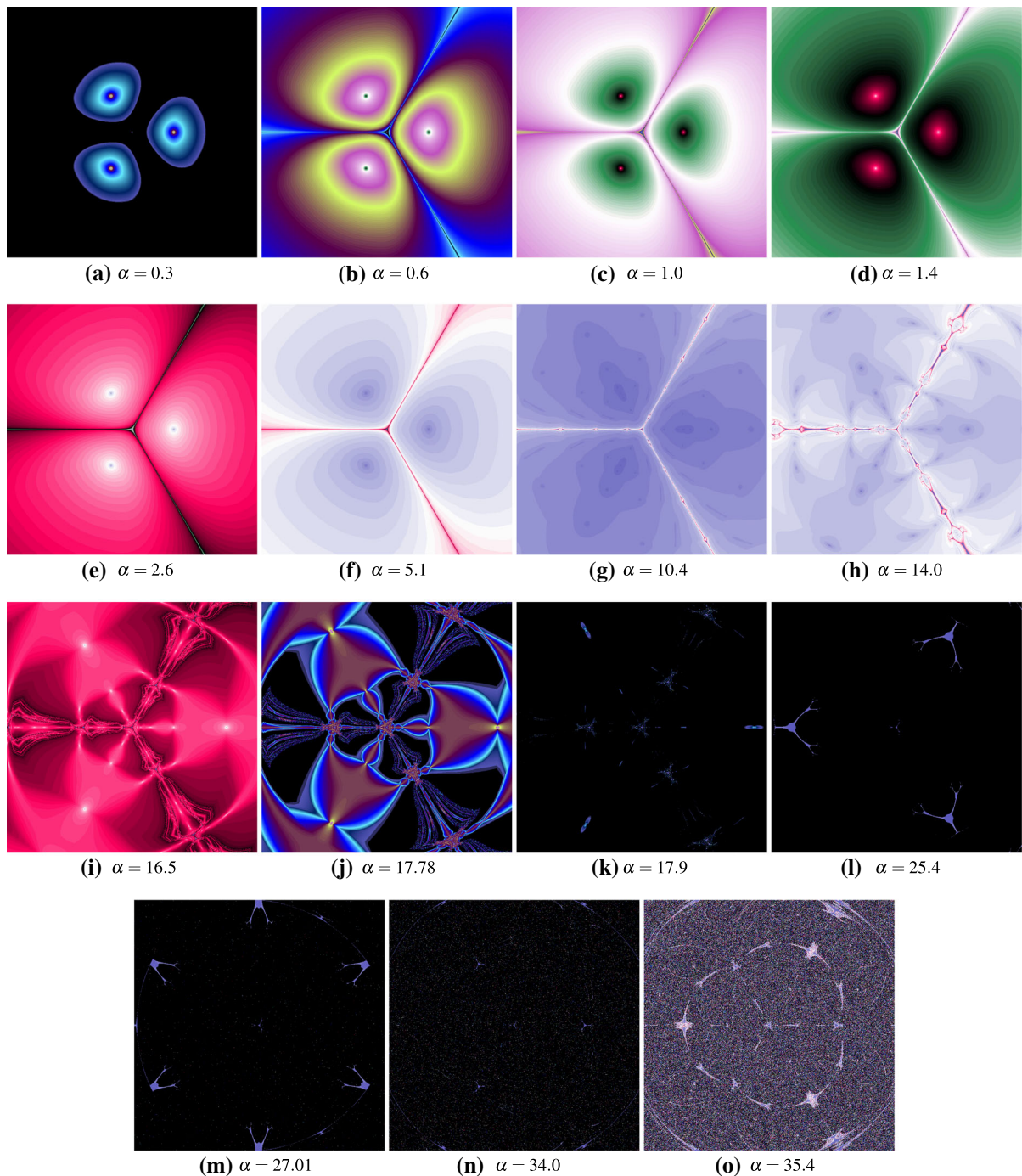




**Fig. 5** Examples of basins of attraction for  $p(z) = z^3 - 1$  for various values of  $\alpha$

points have converged, i.e. CAI equals to 1.0, so the convergence is better than in the case of the RNM. For  $\alpha \in [17.75, 33.74]$ , only a small percentage (more or less 1%) of points have converged. For  $\alpha > 33.74$  the

number of converging points changes in a random way. By observing the time plot, one can see a similar tendency as for ANI. Since each iteration takes nearly the same amount of time, the time plot is in high corre-

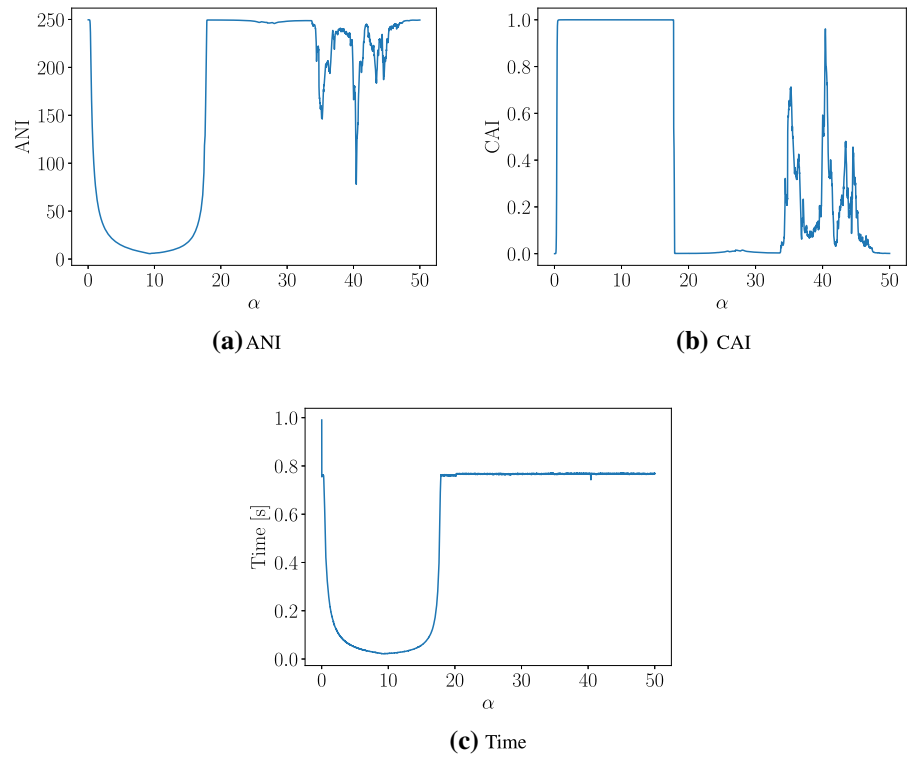


**Fig. 6** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^3 - 1$  for various values of  $\alpha$

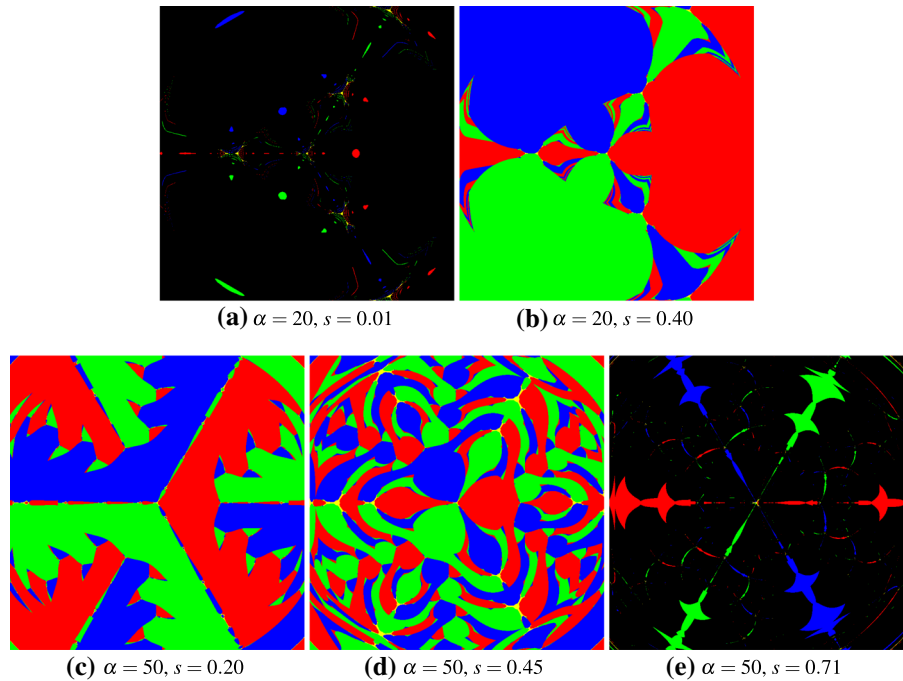
lation with the average number of iterations. When  $\alpha$  increases, the time of computations is getting shorter down to its minimum 0.021 s taken for  $\alpha = 9.29$ . Then, the time is getting longer up to  $\alpha = 17.9$ . For  $\alpha > 17.9$ ,

the time oscillates around 0.76 s. When we compare the times obtained by the M-RNM, especially with the minimal value (0.021 s), to the time obtained by the RNM, i.e. 0.27 s, we see that the M-RNM can generate

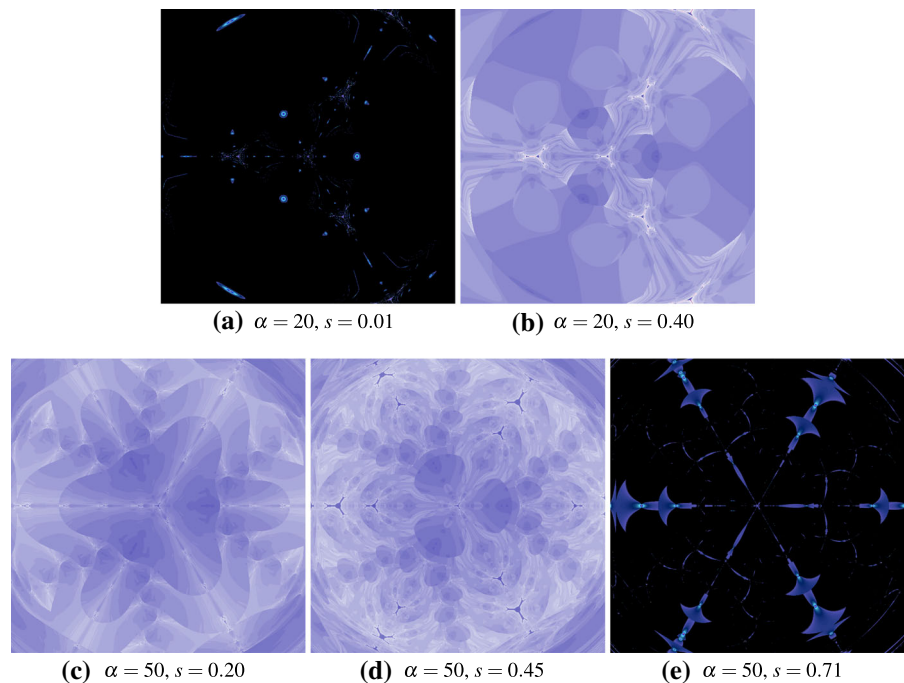
**Fig. 7** Numerical results for  $p(z) = z^3 - 1$  and the various measures



**Fig. 8** Examples of basins of attraction for  $p(z) = z^3 - 1$  for various values of starting parameters  $\alpha$  and various values of  $s$



**Fig. 9** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^3 - 1$  for various values of starting parameters  $\alpha$  and various values of  $s$



the polynomiograph much faster (using less number of iterations). As one can expect, all these plots confirm the visual results presented in Figs. 5 and 6.

#### 4.1.2 Variable $\alpha$ sequence

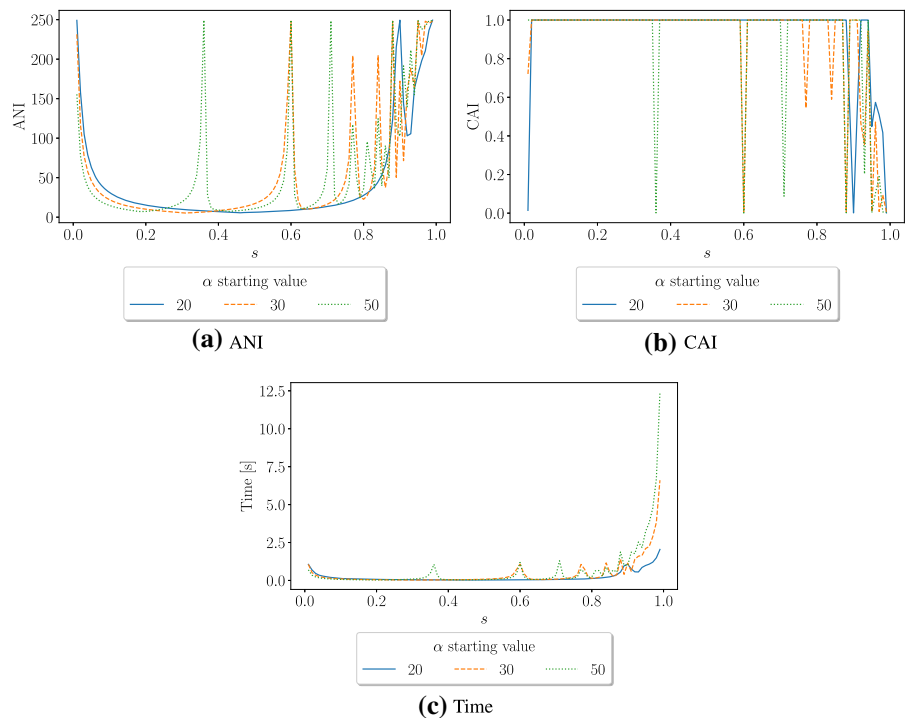
In this subsection, we analyse the M-RMN algorithm for the case of non-constant  $\alpha$  sequence. We take the starting values of  $\alpha$  as 20 and 50, and a discrete scale values  $s$  from the interval  $(0, 1)$ . As a result of the experiments the images of basins of attractions, presented in Fig. 8, the images of dynamics, shown in Fig. 9, for  $z^3 - 1$  and numerical characteristics of ANI, CAI and time generation, given in Fig. 10, were obtained. From the sets of the obtained images, the most representable ones were chosen. In the images in Fig. 8a, e dominates the colour black, what means non-convergence and denotes very bad behaviour of the M-RNM algorithm. The remaining images, from Fig. 8b–d, show that the M-RMN method is convergent. In these images, mostly three colours are clearly seen: red, green, and blue, all denoting basins of attractions, and small yellow areas related to the critical point. The obtained basins look different compared to those generated using the RNM or by the classical Newton's method. They have complicated shapes with rotational symmetry, split into several bigger and smaller areas.

Their structure is not fractal with typical braids as in the classical Newton's method and different to the three triangles creating basins as in the case of the RNM algorithm. Thus then, the M-RMN in the considered cases of variable  $\alpha$  sequences is less robust in comparison with the RNM or M-RNM with constant  $\alpha$  sequences.

For the same choice of  $\alpha$  and  $s$  parameters as for the basins of attraction, in Fig. 9, dynamics of the M-RMN root finding process with variable  $\alpha$  sequence is presented. In Fig. 9a, e dominates the colour black what means the lack of convergence, whereas in Fig. 9b–d bright bluish colour is mostly seen what means, according to the used colour map (Fig. 4), a very low number of iterations, i.e. very high speed of convergence.

In these experiments, we obtained the following best results: ANI = 5.54, CAI = 1.0, time generation = 0.027 s for starting  $\alpha = 20, s = 0.46$  and ANI = 7.34, CAI = 1.0, time generation = 0.040 s for starting  $\alpha = 50, s = 0.43$ . When comparing them to the constant case for  $\alpha = 9.32$ , ANI = 5.62, CAI = 1.0, time generation = 0.021 s, we see similar numerical results in both cases but basins of attraction are very different—they are more complicated in the case of variable  $\alpha$  sequence. More details concerning the M-RNM with non-constant  $\alpha$  sequence can be extracted from the plots shown in Fig. 10. Namely, from Fig. 10a,

**Fig. 10** Numerical results for  $p(z) = z^3 - 1$  and the various measures and the various starting points



b, one can see that the blue lines for starting  $\alpha = 20$  are more regular, whereas the lines for starting  $\alpha = 30$  and even more for starting  $\alpha = 50$  are chaotic. The investigated M-RNM with variable  $\alpha$  sequence has the best behaviour for values of  $s$  parameter taken from the middle of the interval  $(0, 1)$ , whereas when  $s$  is close to 0 or 1, then the algorithm works badly. Moreover, a suitable choice of the starting  $\alpha$  is essential because it influences the shape of the basins of attractions. The work of the considered algorithm seems to be the best (i.e. robust with low ANI and CAI close to one) for starting  $\alpha$  equal to 20.0 and  $s$  close to 0.5.

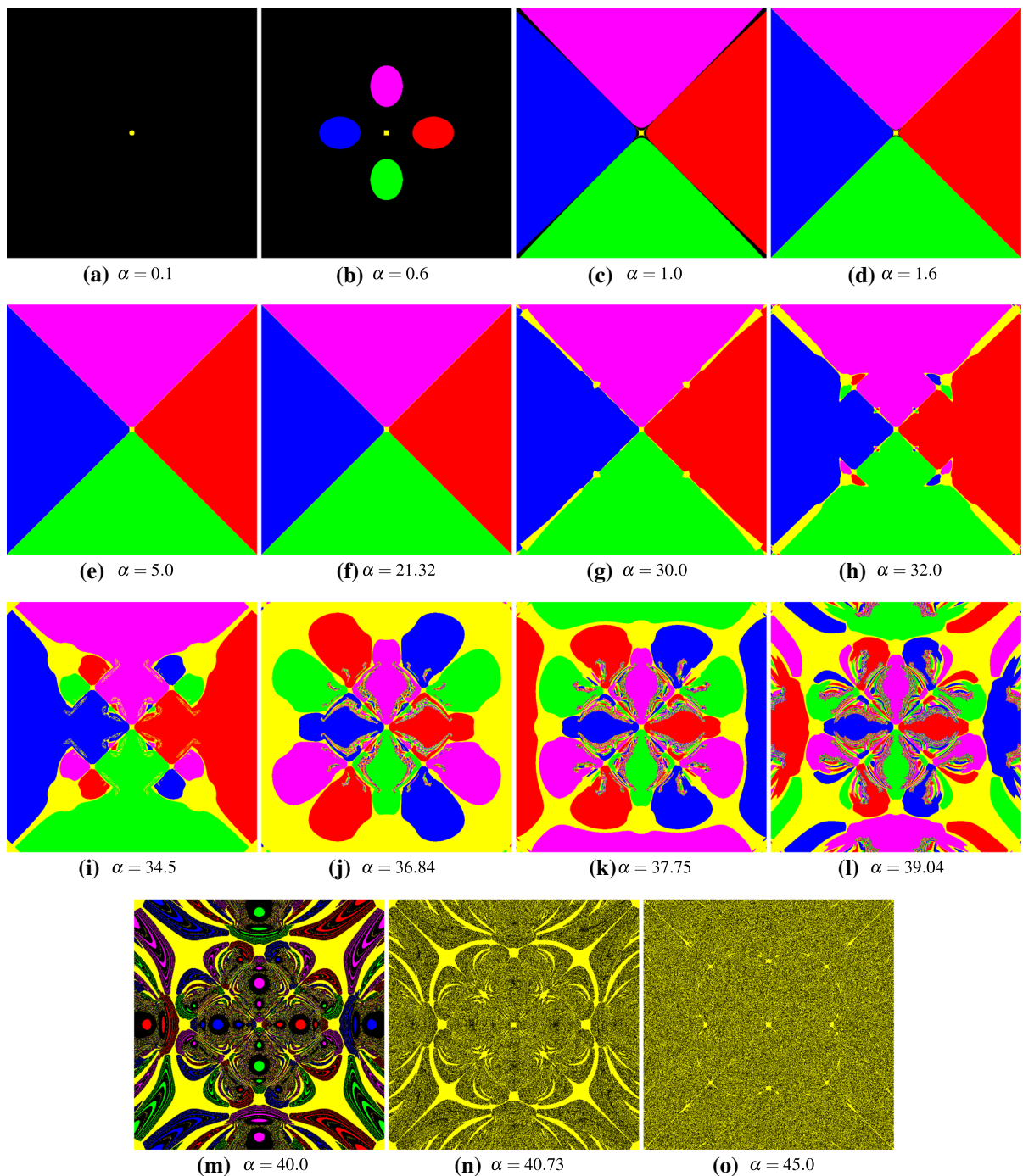
## 4.2 $p(z) = z^4 - 1$

### 4.2.1 Constant $\alpha$ sequence

In Fig. 11, the polynomiographs' sequence showing the evolution of the five basins (four for the roots and one for the critical point  $0 + 0i$ ) of  $z^4 - 1$  depending on  $\alpha$  is given. The basins of the roots are coloured by four colours: red, green, blue, and magenta. Black colour denotes the lack of convergence, whereas yellow—convergence to the critical point. In Fig. 11c–f, four almost triangular basins fill the whole area of poly-

nomiographs except the yellow critical point. When  $\alpha$  is decreasing, starting from  $\alpha = 1.0$  (corresponding to the RNM), the basins became smaller. At  $\alpha = 0.1$  they disappear entirely, and almost all points are not convergent except for the yellow critical one. When the parameter  $\alpha$  is increasing starting from 30.0 to 34.5 on the basins' boundaries gradually appear yellow artefacts that are related to points converging to the critical one and are opening new areas of convergence to the roots inside their old basins. Further increase in  $\alpha$  causes that the new areas of convergence to the roots are getting larger and the area of yellow colour is expanding, whereas the old basins are getting smaller up to their disappearance. The regular symmetric areas visible on the polynomiographs have colour according to some easily understandable scheme. With the further increase in  $\alpha$ , one can observe that areas of yellow colour are gradually replaced by new areas of convergence to roots or the critical point and areas of non-convergence. Finally, for  $\alpha \geq 40.73$ , only convergence to the critical point or non-convergence is observed. In this case, a regular



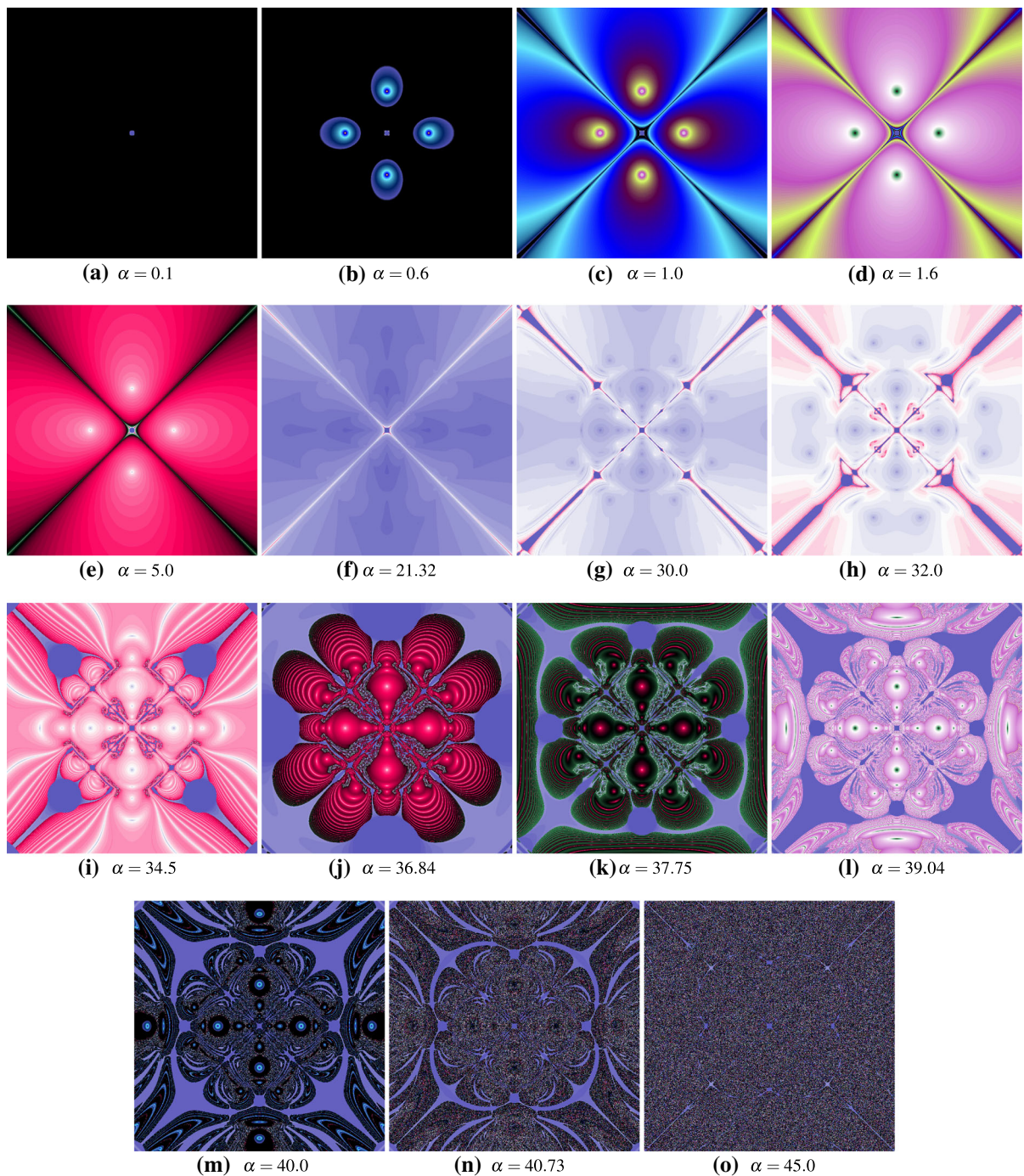


**Fig. 11** Examples of basins of attraction for  $p(z) = z^4 - 1$  for various values of  $\alpha$

pattern occurs or a randomly distributed mix of yellow and black points are visible.

Further information concerning the speed of convergence of the method can be extracted from the poly-

nomiographs presented in Fig. 12 that show dynamics of the considered root-finding process for the same sequence of  $\alpha$  values as in Fig. 11. Figure 12 presents the symmetric reference image for  $\alpha = 1.0$ , i.e. the

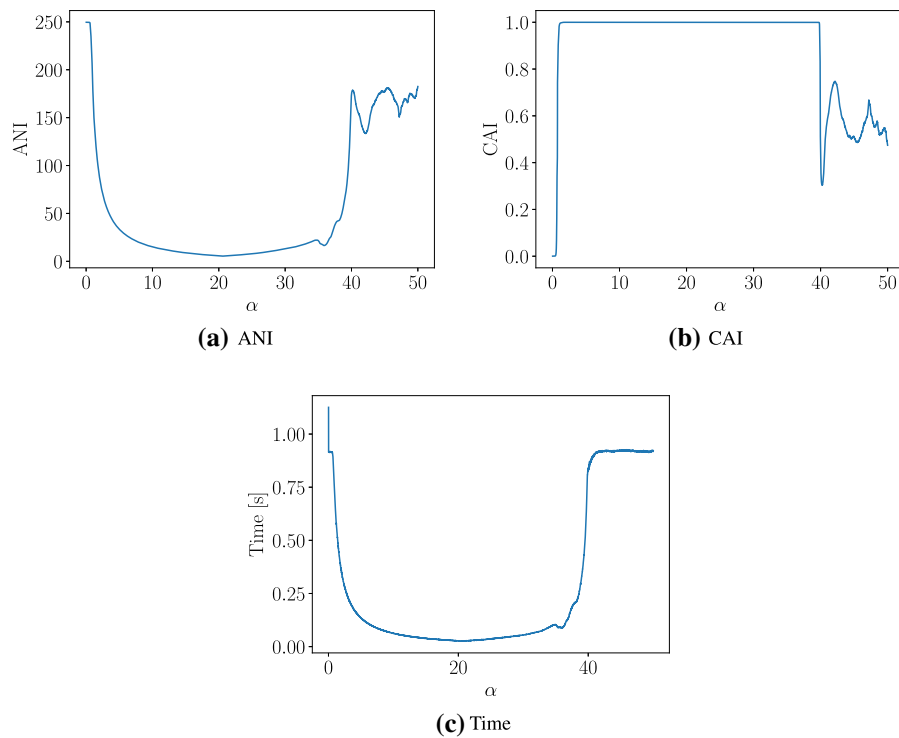


**Fig. 12** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^4 - 1$  for various values of  $\alpha$

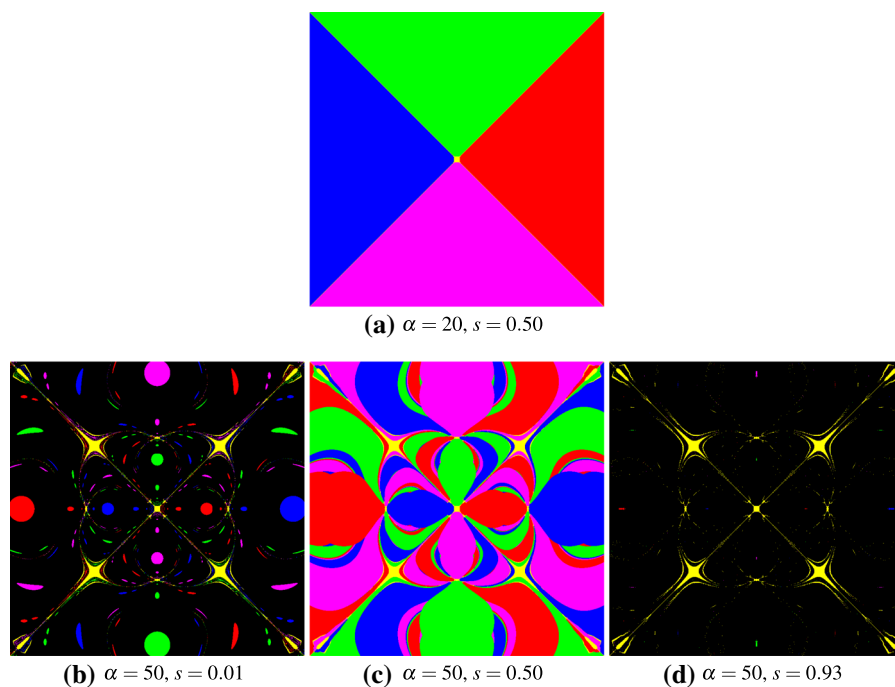
case of the RNM. Four roots and the critical points are well-visible. Yellow, violet, blue, and light blue colours are seen in the image, which means the high number of iterations. No chaos and fractal structures are seen

on the diagonals. If  $\alpha$  is decreasing from 1.0 more and more nonconvergence appears (Fig. 12a, b). When  $\alpha$  is increasing from 1.0 up to about 32.0 the colours in Fig. 12e–h are moving to the left part of the colour

**Fig. 13** Numerical results for  $p(z) = z^4 - 1$  and the various measures

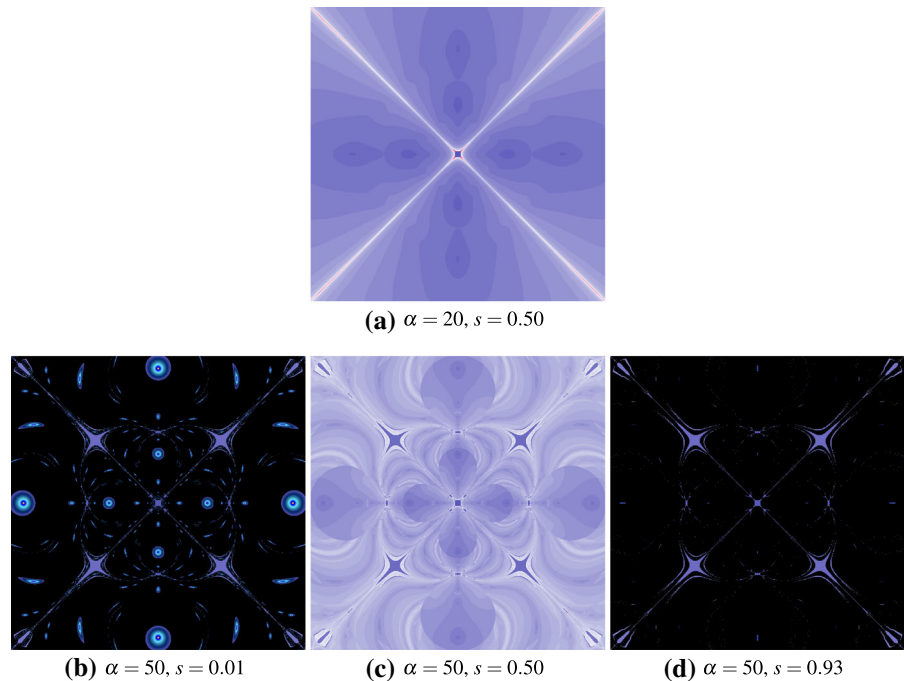


**Fig. 14** Examples of basins of attraction for  $p(z) = z^4 - 1$  for various values of starting parameters  $\alpha$  and various values of  $s$





**Fig. 15** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^4 - 1$  for various values of starting parameters  $\alpha$  and various values of  $s$



map (Fig. 4) what means that the number of iterations is going down. Further increase of  $\alpha$ , as in Fig. 12i–l, reveals intriguing symmetric patterns coloured by colours in the left part of the colour map (Fig. 4) that is related to a rather relatively small number of iterations. For  $\alpha \geq 40.0$  (Fig. 12m–o) more and more random points are observed, and a large number of iterations is performed what is coded by the colours used in the right part of the colour map.

The plots of ANI, CAI and time generation (in seconds) versus  $\alpha$  are presented in Fig. 13. This completes the analysis of the considered root-finding method in the case of  $z^4 - 1$ . Observe that for  $\alpha = 1.0$  (RNM) the following values of parameters were obtained: ANI = 179.9, CAI = 0.98, time = 0.68 s, whereas for  $\alpha = 20.47$  ANI parameter obtained minimal value 5.41 when CAI = 1.0 and time = 0.025 s. This means that the M-RNM for  $\alpha = 20.47$  has attained the best improvement in comparison to the RNM—the number of iterations was reduced 33 times and time reduction was 27 times with CAI close to 1.0. Similar behaviour occurs on the flat parts of ANI and CAI plots for  $\alpha$  values roughly in the range [10, 30]. When  $\alpha > 40.0$  random fading oscillations are seen, what means that the M-RNM loses its stability and stops working properly.

Summing up, the M-RNM improves the RNM in the number of iterations essentially in the case of polynomial  $z^4 - 1$  for  $\alpha = 20.47$ . Less improvement is observable for  $\alpha$  values less or greater than the optimal one being  $\alpha = 20.47$ . Both algorithms have global convergence. The M-RNM algorithm does not lose its robustness for parameter  $\alpha$  varying in a relatively wide range of  $\alpha$ . We cannot see typical fractal braids that are observed for polynomiographs generated via classical Newton's method.

#### 4.2.2 Variable $\alpha$ sequence

In this subsection, we show some examples of the M-RMN for variable  $\alpha$  sequence. As in Sect. 4.1.2, we fixed the starting values of  $\alpha$  as 20 and 50 since they are the most representative. In Fig. 14, the examples of basins of attraction for the considered polynomial are presented for these starting values of  $\alpha$  and different values of parameter  $s$ . As in the previous polynomial case, the values of  $s$  were chosen to give the most diversity of this presentation. It is worth mentioning that in the case of starting  $\alpha = 20$  for all values of  $s$  the polynomiograph looks the same, whereas for starting  $\alpha = 50$ , it changes from the one with much non-converging points (black areas, Fig. 14b) via the lack of them (Fig. 14c) and back again (Fig. 14d).

Similarly as in the previous subsection, in Fig. 15, the dynamics polynomiographs are presented for different starting values of  $\alpha$  and  $s$  (the same as in Fig 14). For starting  $\alpha = 20$ , the dynamics remains the same for all values of  $s$ . On the other hand, for starting  $\alpha = 50$  we can see the diversity of convergence for different values of  $s$ .

Finally, in Fig. 16, the plots of ANI, CAI and generation time are presented for the three different starting values of  $\alpha$ . As one can see for starting  $\alpha$  equal to 20 and 30 the number of iterations is constant and so the convergence area index and time of generation. In the case of starting  $\alpha = 50$ , we can observe some peaks in the plots. These peaks represent the polynomiographs with many non-convergent points. The higher the peak, the more the non-convergent points. Moreover, we can observe that the best choice of  $s$  is about 0.4 since it assures the shortest generation time and the best convergence for all tested starting values of  $\alpha$ . In more detail, for the variable  $\alpha$  sequence the shortest generation times were obtained as follows: for  $\alpha = 20$ , the time was oscillating around 0.033 s, for  $\alpha = 30$ , the time was oscillating around 0.067 s, and for  $\alpha = 50$ , the shortest time was 0.040 s.

### 4.3 $p(z) = z^5 + z$

#### 4.3.1 Constant $\alpha$ sequence

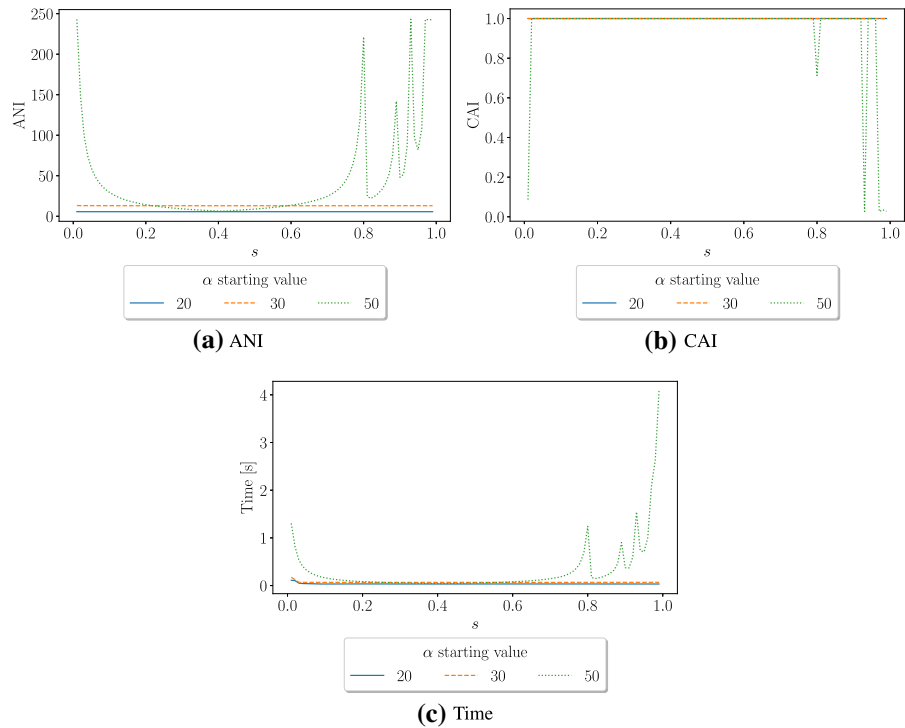
In the last example, we will consider the polynomial  $p(z) = z^5 + z$ . It has five distinct roots and four critical points, i.e.  $-0.4728 + 0.4728i$ ,  $-0.4728 - 0.4728i$ ,  $0.4728 - 0.4728i$ ,  $0.4728 + 0.4728i$ . The basins of attraction obtained for this polynomial are presented in Fig. 17. The roots are coloured by red, green, blue, cyan, and magenta colours. Firstly, let us look at the basins of attraction for the RNM with the Picard iteration (Fig. 17c). We can easily observe that many of the points have not converged to any root or critical point. Most of the converging points are converging to one of the roots, namely to  $0 + 0i$  (cyan colour). For the Picard iteration, we can increase the number of converging points by simply increasing the value of  $M$  obtaining convergence of all points in the considered area for  $M = 1255$ , which is a high value. If we decrease the value of  $\alpha$  in the Mann iteration below 1.0, then we can observe that the M-RNM converges to only one of the roots and that the basins for this

root shrunk. Thus, the method behaves worse than the RNM. If we increase the value of  $\alpha$  above 1.0, then we see, the opposite behaviour, i.e. the non-convergent points for  $\alpha = 1.0$  start to converge to the four roots other than  $0 + 0i$ . For  $\alpha = 3.0$ , we see almost full coverage of the area with the convergent points (see Fig. 17e), but for  $\alpha = 10.0$ , we obtain the situation in which all points converge (see Fig. 17f). We obtain this by using  $M = 250$  in the Mann iteration, which is considerably smaller than  $M = 1255$  for the Picard iteration. By increasing  $\alpha$  further, we can observe that some points lose convergence to the root represented by the cyan colour. For  $\alpha = 22.0$ , we obtain the situation in which none of the points converges to the  $0 + 0i$  root. By increasing  $\alpha$  further, we see that some chaotic behaviour occurs. We still see some visible basins of attraction for the roots, but they are much smaller than for the lower values of  $\alpha$ . Most of the points converge to a random root.

When we look at the dynamics presented in Fig. 18, we can observe a very interesting behaviour of the M-RNM. Similarly to the basins of attraction, we start with the polynomiograph for the RNM with the Picard iteration (Fig. 18c). From this polynomiograph, we can observe that very fast convergence is obtained near the  $0 + 0i$  root. Moreover, the farther away from the root, the more iterations the method needs to perform to reach this root. When we decrease the value of  $\alpha$  in the Mann iteration below 1.0, the speed of convergence is also decreasing, i.e. we need to perform more iterations to find the root. The number of iterations—depicted by colour—changes radially from the  $0 + 0i$  root. If we take values of  $\alpha$  greater than 1.0, then we can observe a considerable reduction in the number of performed iterations. For instance, in the central part, we see a more dark pink colour, which means that to reach the root, we need fewer iterations than in the case of  $\alpha = 1.0$ . In the same time, we can observe that the dynamics increases. From Fig. 18f, we see that the M-RNM method needs a very small number of iterations to find the roots (we see many points with dark pink and light violet colours). However, when we increase the value of  $\alpha$  further, we see that in the cross-like area in the centre of the polynomiograph, the number of iterations increases and the method loses its convergence (black colour represents the maximal number of iterations), see Fig. 18h. After a certain threshold, the M-RNM starts to behave increasingly chaotic. In Fig. 18i–o, we see this behaviour. Moreover, we can



**Fig. 16** Numerical results for  $p(z) = z^4 - 1$  and the various measures and the various starting points

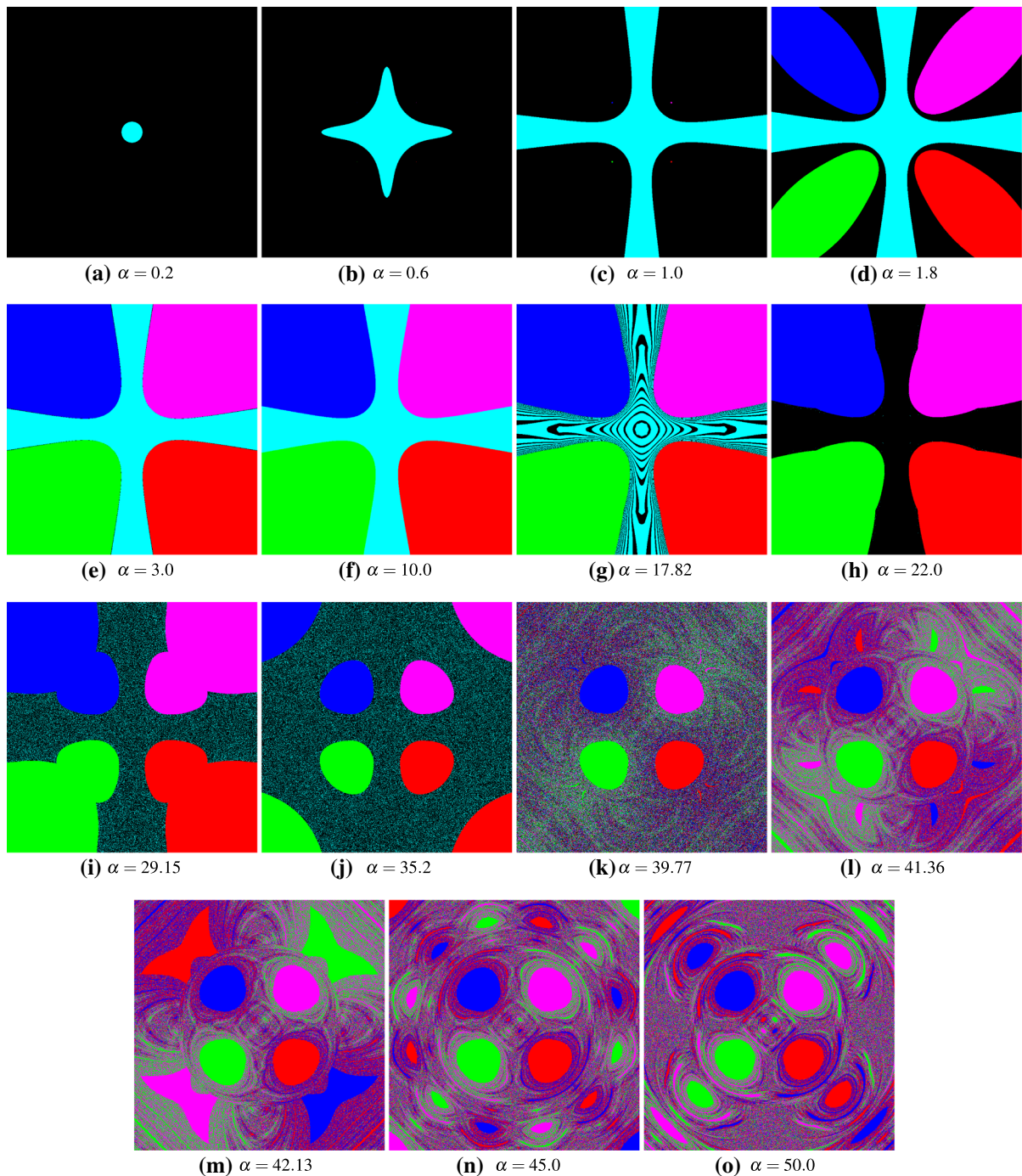


observe that for the points that converge a similar number of iterations is needed to find the roots, i.e. we see light violet and pink colour, which means that we need less than half of the maximal number of iterations ( $M = 250$ ).

The plots of ANI, CAI and polynomiograph generation time (in seconds) for  $p(z) = z^5 + z$  are presented in Fig. 19. The values of these measures for the RNM with the Picard iteration ( $\alpha = 1.0$  in the Mann iteration) are equal to 229.77, 0.20 and 1.02 s, respectively. When we look at the ANI plot (Fig. 19a), we see that for values of  $\alpha$  less than 1.0 the value of ANI increases its value obtaining the maximal value of 250 for  $\alpha < 0.07$ . For values of  $\alpha$  greater than 1.0, we see that ANI decreases with the increase in  $\alpha$ . The local minimal value of 25.23 is attained at  $\alpha = 15.59$ . Then, ANI increases obtaining the local maximal value of 197.11 for  $\alpha = 39.46$ . For  $\alpha > 39.46$ , we can observe another decrease in ANI. The minimal value of 23.92 (which is the global minimum) is attained at  $\alpha = 48.71$ . By comparing the results for  $\alpha > 1.0$  with the result obtained for the RNM (229.77) we see that for all the values of  $\alpha$  we get a lower value of ANI. Now, let us look at the CAI plot presented in Fig. 19b. From the plot, we see that for  $\alpha < 1.0$  we obtain results worse than for the RNM.

The lowest value of CAI equal to 0.0 is obtained for  $\alpha < 0.07$ . If we change  $\alpha$  in the opposite direction, i.e. we increase its value above 1.0, we see that the value increases violently obtaining the maximal value of 1.0 for  $\alpha = 4.16$ . The value 1.0 of CAI does not change until  $\alpha = 17.79$ , when it starts to decrease. The local minimum of 0.31 is attained at  $\alpha = 39.46$ . Then, again the value of CAI starts to increase obtaining the value near 1.0 ( $\approx 0.99$ ) for  $\alpha > 40.7$ . Additionally, when we compare the results for  $\alpha > 1.0$ , then we see that the obtained results for the M-RNM are better than in the case of the RNM (0.20). In the last plot, presented in Fig. 19c, we see the polynomiographs' generation times. The overall tendency observed in this plot is very similar to the one obtained for ANI (Fig. 19a). For  $\alpha < 1.0$  we get longer times than in the case of the RNM, obtaining the longest time 1.27 s for  $\alpha = 0.01$ , whereas for  $\alpha > 1.0$  times get shorter. The visible local extrema of 0.12 s, 1.02 s and 0.324 s are attained at  $\alpha = 15.81$ , 39.44 and 48.42, respectively. The shortest time of 0.12 s is considerably shorter than the time for the RNM (1.02 s).

One interesting thing that we can observe by comparing the plots for ANI (Fig. 19a) and times (Fig. 19c) is that the value of  $\alpha$  for which we get the lowest value

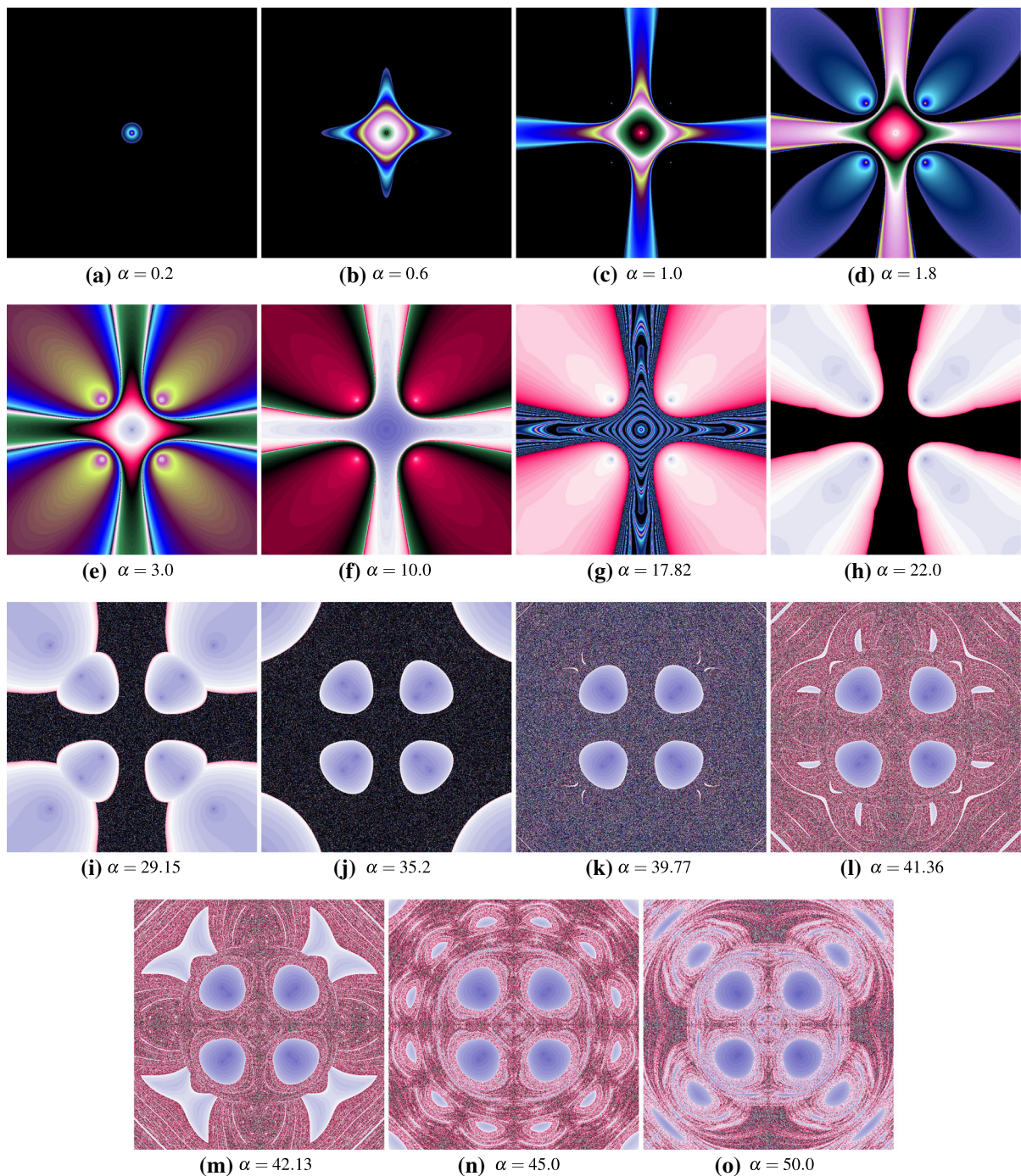


**Fig. 17** Examples of basins of attraction for  $p(z) = z^5 + z$  for various values of  $\alpha$

of ANI ( $\alpha = 48.71$ ) does not coincide with the value for the shortest time ( $\alpha = 15.81$ ). Moreover, when we compare the CAI values for  $\alpha = 48.71$  (0.99) and  $\alpha = 15.81$  (1.0), we see that in this case, the lowest

value of ANI does not guarantee the best convergence in the area, i.e. some points for  $\alpha = 48.71$  do not converge to any root, whereas for  $\alpha = 15.81$ , all points have converged. Therefore, when we want to select the





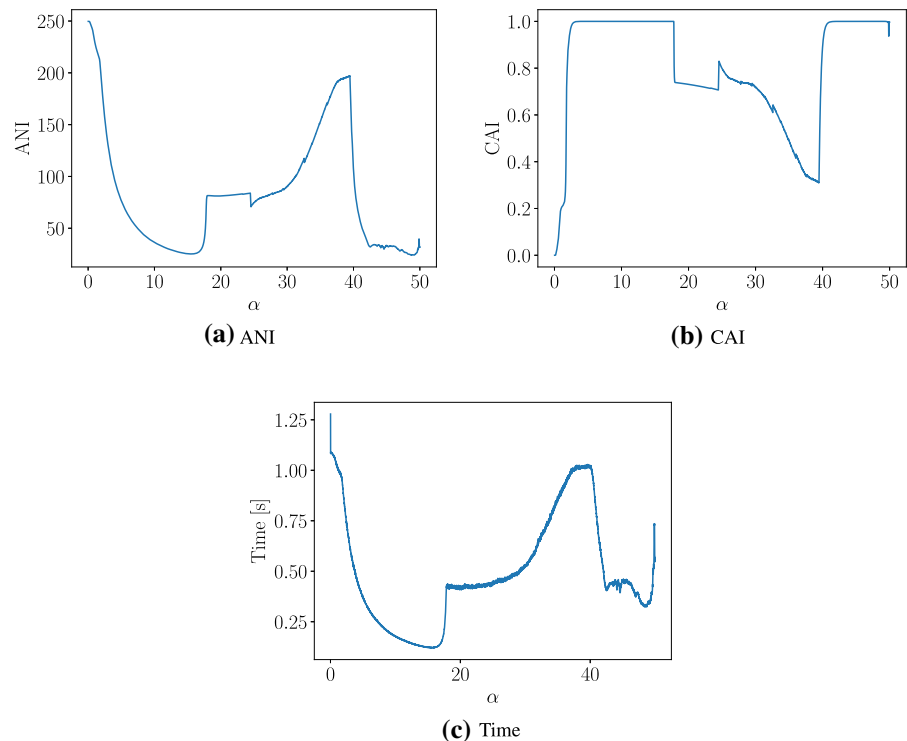
**Fig. 18** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^5 + z$  for various values of  $\alpha$

best value of  $\alpha$  that assures good convergence and performance, we need to consider all three measures (ANI, CAI, time) simultaneously.

#### 4.3.2 Variable $\alpha$ sequence

In this subsection, we present some examples obtained for the M-RMN with a variable  $\alpha$  sequence. The

**Fig. 19** Numerical results for  $p(z) = z^5 + z$  and the various measures



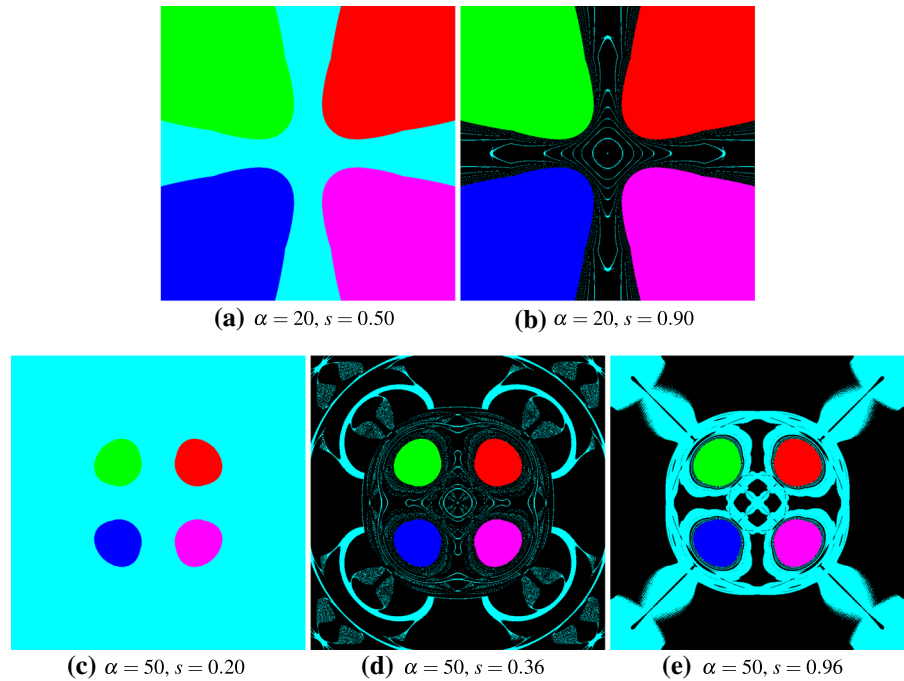
graphical examples—like in the case of the previous polynomials—were generated for two starting values of  $\alpha$ , namely 20 and 50, and variable  $s$ . The obtained basins of attraction are presented in Fig. 20. For the starting  $\alpha = 20$  (see Fig. 20a, b), the basins of attraction for the majority of the  $s$  values look like the basins presented in Fig. 20a. The method converges to all roots, and there are no visible non-converging points. The loss of convergence of the method appears for the  $s$  values near 1, where the method loses its convergence in the cross-like region, see Fig. 20b. For the other starting  $\alpha$  value (see Fig. 20d, e), the basins of attraction look different, i.e. we do not see the cross-like region. However, also in this case, for the majority of the  $s$  values, the basins look like the ones presented in Fig. 20c. For values of  $s$  near 1 and for some single values of  $s$  in the rest of the  $(0, 1)$  interval, e.g. for  $s = 0.36$ , we lose convergence of the method, see Fig. 20d, e.

The polynomiographs showing the dynamics and the speed of convergence for the same values of the parameters (starting  $\alpha$  and  $s$ )—like in the case of basins of attraction—are presented in Fig. 21. When we look at starting  $\alpha = 20$ , we see that the dynamics and the speed of convergence change only in the cross-like region. In the other regions, the polynomiographs do

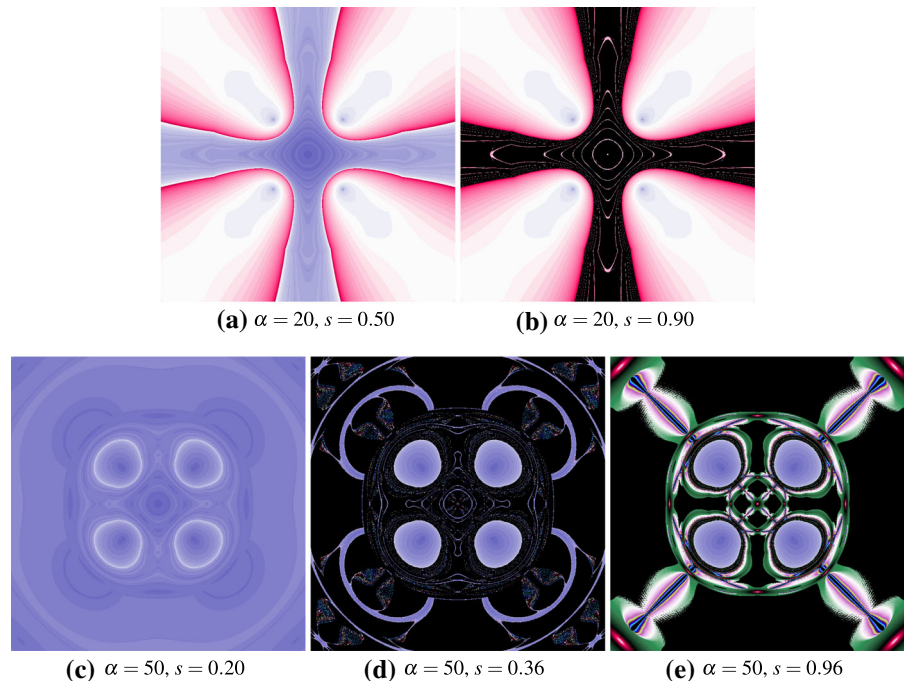
not change. The speed of convergence, compared to the RNM (Fig. 18c), is faster for the majority of the  $s$  values. Only for values of  $s$  near 1 the speed in the cross-like region is worse. For starting  $\alpha = 50$ , we see that for the polynomiographs with good convergence (Fig. 21c) the speed of convergence is very fast. For the points that converge in the other cases (Fig. 21d, e), the speed of convergence is faster or comparable to the RNM. Moreover, when we compare the polynomiographs for starting  $\alpha = 50$ , and the one for a constant  $\alpha = 50$  (Fig. 18o), we see that in the variable case there is no chaotic behaviour that is visible for the constant case.

The plots of ANI, CAI and the generation time for three starting values of  $\alpha$ , namely 20, 30 and 50, are presented in Fig. 22. From the ANI plot (Fig. 22a), we can see that in all three cases the method obtains larger values of ANI for  $s$  near 0 and 1. Moreover, for starting  $\alpha$  values equal to 30 and 50 we observe some peaks, which correspond to the exemplary basins of attractions presented in Fig. 20d. When we compare the obtained results with the result for the RNM, i.e. 229.77, we see that for the variable  $\alpha$  sequence, we obtain much lower values of ANI. Only in the peaks visible for starting  $\alpha = 50$ , we can see the values close to 229.77.

**Fig. 20** Examples of basins of attraction for  $p(z) = z^5 + z$  for various values of starting parameters  $\alpha$  and various values of  $s$



**Fig. 21** Examples of the speed of convergence and dynamics polynomiographs for  $p(z) = z^5 + z$  for various values of starting parameters  $\alpha$  and various values of  $s$



The minimal values of ANI—for the three considered starting  $\alpha$  values—equal to 16.298, 9.789 and 3.671 were attained for  $s$  equal to 0.45, 0.3 and 0.18, respectively. By looking at the plots of CAI (Fig. 22b), we can observe that for most of the  $s$  values the value of CAI is

equal to 1.0. The values less than 1.0 are only obtained for  $s$  close to 1 and in the visible peaks that appear in the same points as the peaks in the ANI plot. By comparing these results with the value of CAI obtained for RNM, i.e. 0.20, we can see that the convergence of M-RNM



with the variable  $\alpha$  sequence is much better. From the last plot (Fig. 22c), i.e. the generation time plot, we can observe a very similar tendency as in the case of the ANI plot. The only difference is that for the values of  $s$  near 1 we can observe a very large time difference compared to the other peaks. The best times in all three considered cases are much shorter than in the case of the RNM (1.02 s), namely for  $\alpha = 20$  the time was 0.1 s (attained at  $s = 0.45$ ), for  $\alpha = 30$  the time was 0.063 s (attained at  $s = 0.3$ ), and for  $\alpha = 50$  the time was 0.026 s (attained at  $s = 0.18$ ).

#### 4.4 Comments

We only consider positive values of parameter  $\alpha$  in the M-RNM algorithm because for negative values of  $\alpha$  convergence cannot be obtained. For  $\alpha < 1$ , when  $\alpha$  is getting smaller and smaller, the M-RNM behaves increasingly badly—more and more points of non-convergence appear in polynomiographs. The M-RNM behaves better than the RNM for  $\alpha > 1$ , i.e. more points converge, and the number of iterations is smaller.

The optimal value of  $\alpha$  is such that the ANI measure is as low as possible, and the CAI index simultaneously is as close as possible to 1. Such cases occur for  $z^3 - 1$  and  $z^4 - 1$  when  $\alpha$  attain 9.32 and 20.47 for which ANI measure attain global minimum 5.63 and 5.41, respectively. For other polynomials, the optimal value of  $\alpha$  can be taken as the one for which ANI measure attains local minimum and CAI index is close to 1. Such a situation occurs for polynomial  $z^5 + z$ . That example shows that because of mutual nonlinear dependency between parameters ANI, CAI and time versus  $\alpha$  it is not possible to formulate general recommendations on how to choose optimal  $\alpha$  to achieve the best improvement of the M-RNM algorithm. However, the best values of  $\alpha$  for a specific polynomial could be determined experimentally via analysing of ANI, CAI and time plots.

From the examples for the variable  $\alpha$  given by Algorithm 3, we can observe that also, in this case, we cannot give general recommendations on the choice of the starting  $\alpha$  and  $s$  values. The relationship between these parameters is also nonlinear, but one should avoid the values of  $s$  near 0 and 1 because for these values the method obtains worse results than for the other values in  $(0, 1)$ .

Similarly, like in the case of the RNM, the boundaries between basins of attraction for M-RNM are sharp

in a wide range of  $\alpha$  values for all considered polynomials, and this method is stable. However, after some threshold value (different for each of the polynomials), the basins of attraction look chaotic and the method loses its stability.

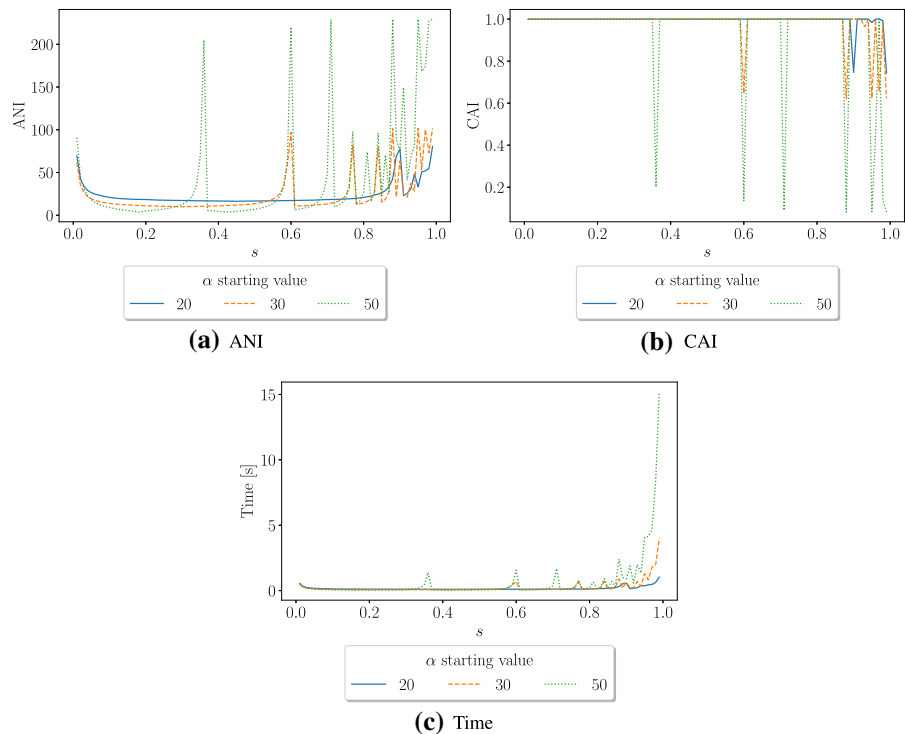
## 5 Artistic patterns from dynamics of the M-RNM

Dynamics of discrete dynamical systems, besides its essential meaning in the analysis of these systems, can also have artistic applications. Graphical presentation of the phase portraits of many discrete dynamical systems can reveal artistic patterns of aesthetic value. We can find many examples of such patterns in the literature, e.g. Ouyang et al. [25] presented spiral patterns that are coloured by the dynamics of the dynamical system that is compatible with some symmetry group, Lu et al. [20] and Chung et al. [9] proposed methods for wallpaper patterns creation from dynamics, and patterns from the dynamics of combined root-finding methods were presented by Gdawiec [12].

When we look at the dynamics of polynomiographs generated by the RNM, e.g. Figs. 6c, 12c or 18c, we can see rather boring patterns—from the artistic point of view. However, when we look at the patterns generated by the Mann iteration in conjunction with the RNM method (see, for example, Fig. 12), we can see that very interesting patterns can emerge. The images presented in Sects. 4.1.1, 4.2.1 and 4.3.1 were obtained for constant sequences of  $\alpha_i$ . In the Mann iteration, we are not limited to only constant sequences, especially if we are interested in artistic applications. Therefore, in the rest of this section, we will introduce some non-constant sequences that will generate very interesting patterns from the dynamics of the M-RNM.

Before presenting examples of artistic patterns, we need to consider two technical subjects that greatly influence the appeal of the patterns presented in the polynomiographs. Let us look at Fig. 23a which was obtained for  $p(z) = z^5 + z$ ,  $\alpha = 17.7$ ,  $M = 200$ ,  $A = [-3, 3]^2$  and  $\varepsilon = 0.001$ . When we look closely at the cross pattern, we see many unpleasant artefacts that look like jagged shape, noise or chaos. This unpleasant effect is caused by the fact that the image has a discrete nature. We sample the signal with a high frequency, i.e. we discretize the space that contains many tiny details that cannot be captured by the discretization process. This is a well-known problem in computer graphics,

**Fig. 22** Numerical results for  $p(z) = z^5 + z$  and the various measures and the various starting points



and it is called aliasing effect [2]. To reduce this effect, one can use one of the many methods of anti-aliasing. The simplest anti-aliasing method is supersampling [2]. In this method, we render the image at a higher resolution than the image to be displayed, usually 2, 4 or 8 times higher. And then, we average the colours in non-overlapping blocks. The block size depends on the resolution's magnification factor, i.e. if  $f$  is the factor, then the block size is equal to  $f \times f$ . The result of using anti-aliasing method to the pattern from Fig. 23a is presented in Fig. 23b, c. We see that the appeal of the pattern in the areas with high frequencies is much better. Now, the pattern does not look noisy or chaotic. In all examples presented in this section, we will use this anti-aliasing method.

The other technical subject that we need to consider when we want to use polynomiographs generated by the proposed M-RNM is choosing a proper colour map. In Sect. 4 in the polynomiographs with iteration colouring, we used only one colour map. However, when we think about artistic applications, we can use various colour maps—even for the same polynomiograph—to obtain very interesting results. In Fig. 24, we see an example of the same polynomiograph ( $p(z) = z^3 - 1$ ,  $\alpha = 17$ ,  $M = 200$ ,  $A = [-3, 3]^2$  and  $\varepsilon = 0.001$ ), but

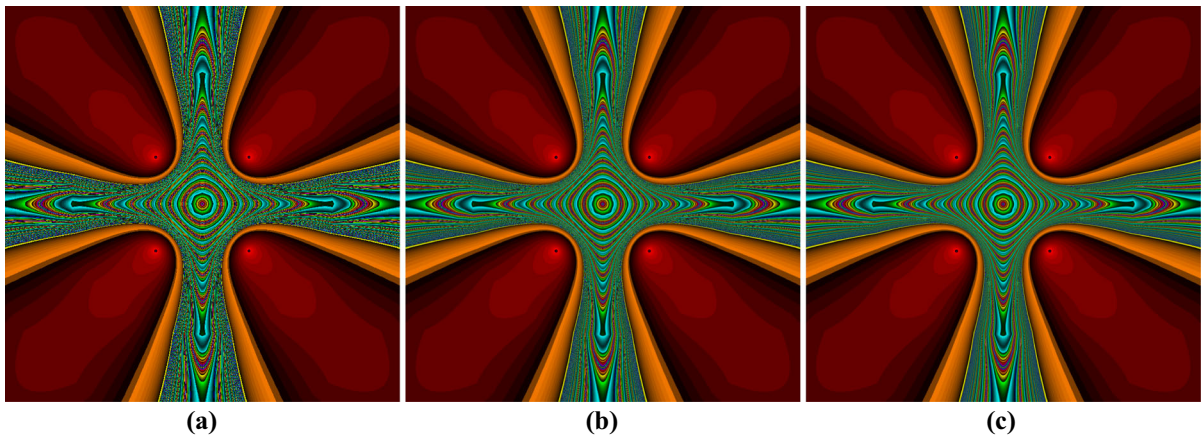
coloured with various colour maps. From the images, we can see that by a proper choice of the colour map we can, for example, emphasize different parts of the same pattern (e.g. see Fig. 24b, e), show banding effect, or hide it (e.g. see Fig. 24a, f). Moreover, the proper colours can create the depth, movement, mood and harmony of an image [29, 32]. Therefore, in the all presented examples in this section, we use various colour maps that greatly increase the polynomiographs' artistic values.

The first sequence that we will consider is very simple, and it is based on the switching of several constant values. Let  $m \in \mathbb{N}$ . Then, we define  $\alpha_i$  in the following way:

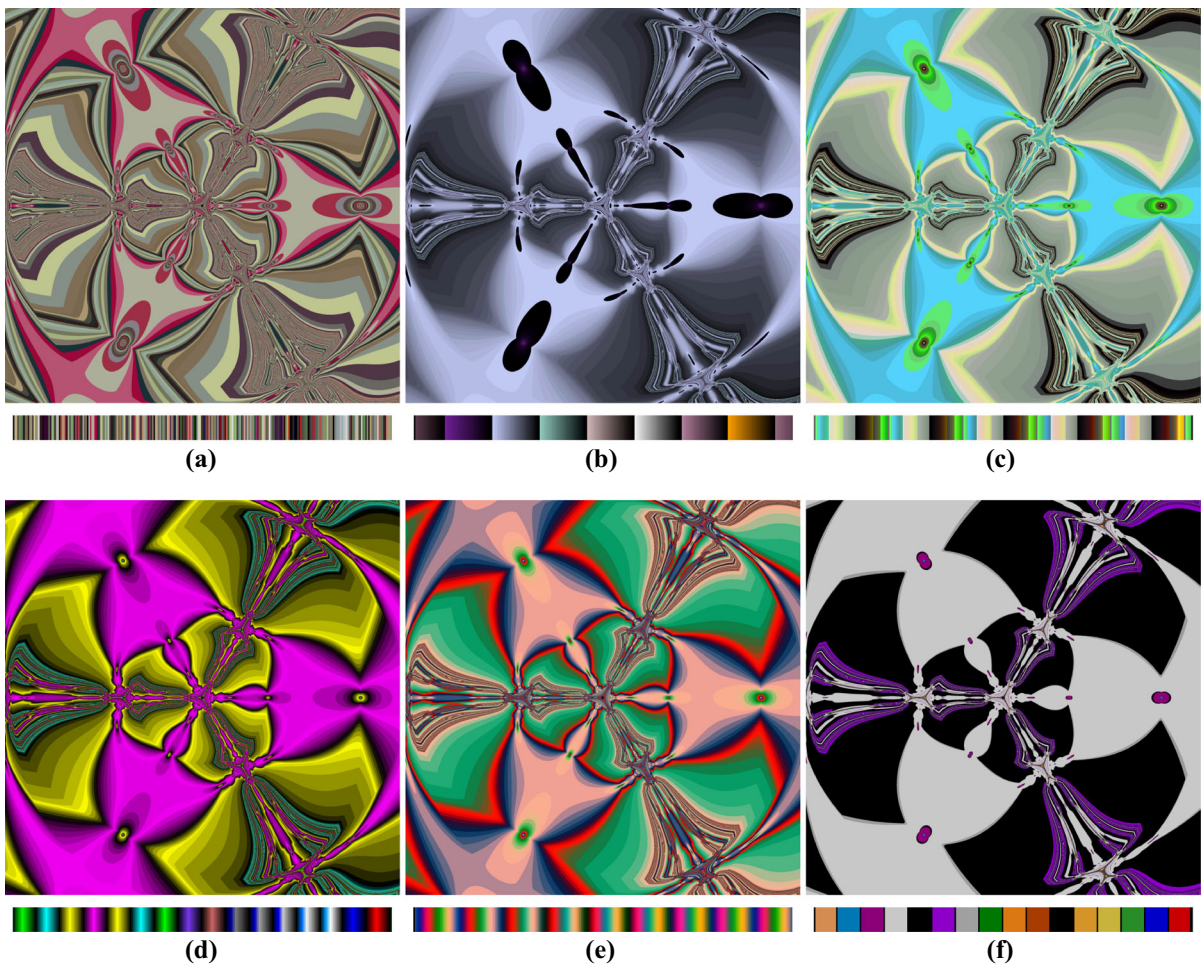
$$\alpha_i = \begin{cases} q_0, & \text{if } i \bmod m = 0, \\ q_1, & \text{if } i \bmod m = 1, \\ \dots & \\ q_{m-1}, & \text{if } i \bmod m = m-1, \end{cases} \quad (20)$$

where  $q_0, q_1, \dots, q_{m-1} \in \mathbb{R}$ .

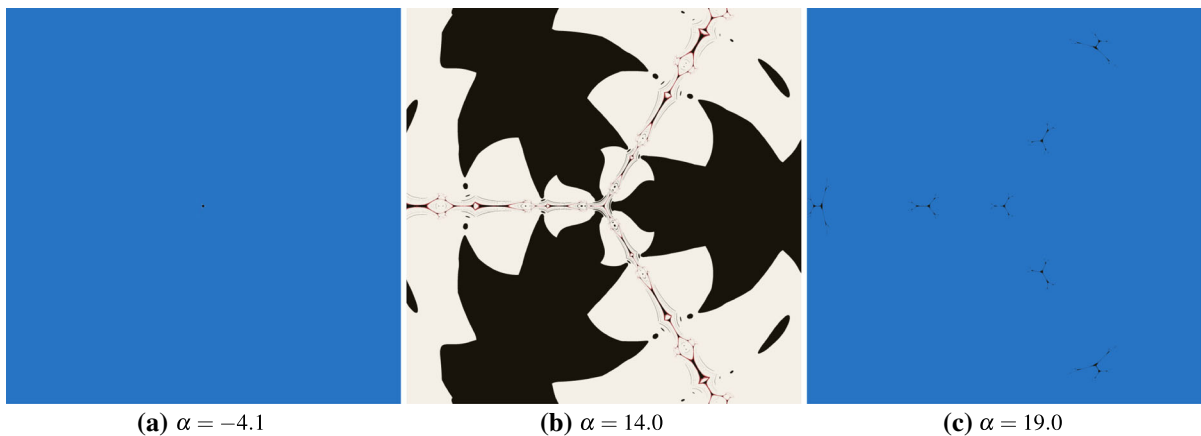
As an example of the use of (20) let us take  $m = 3$  and three constant values  $-4.1$ ,  $14.0$  and  $19.0$ . As we mentioned in our discussion in Sect. 3, negative values of  $\alpha$  cause that the method moves the point from the previous iterations in the opposite direction, than the



**Fig. 23** Polynomiographs of  $p(z) = z^5 + z$  generated without anti-aliasing **a** and with anti-aliasing for two different magnification factors: **b** 2, **c** 4



**Fig. 24** Examples of the same polynomiograph coloured with various colour maps. The colour map is presented below the polynomiograph. (Color figure online)



**Fig. 25** Dynamics polynomiographs for  $p(z) = z^3 - 1$  generated with the M-RNM and various values of  $\alpha$

reduction in the modulus. If we were interested in the convergence to the roots, then it would be undesirable. However, in this section, we are interested in obtaining artistic patterns, so such values are permissible. In our example, we generate dynamics polynomiographs for  $p(z) = z^3 - 1$ ,  $A = [-3.5, 3.5]^2$ ,  $M = 200$  and  $\varepsilon = 0.001$ . Firstly, let us see how the polynomiographs look like for the three values of  $\alpha$ . In Fig. 25, we present these polynomiographs. From these three images, we can see that for  $\alpha = -4.1$  and  $19.0$  the convergence is poor and the patterns are not attractive. In the case of  $\alpha = 14.0$ , the situation is better, but still, the pattern is not attractive from the artistic point of view.

Now, let us use sequence (20) with the three values  $-4.1, 14.0, 19.0$  in various combinations:

- (a)  $q_0 = -4.1, q_1 = 19.0, q_2 = 14.0$ ,
- (b)  $q_0 = 19.0, q_1 = -4.1, q_2 = 14.0$ ,
- (c)  $q_0 = -4.1, q_1 = 14.0, q_2 = 19.0$ ,
- (d)  $q_0 = 14.0, q_1 = -4.1, q_2 = 19.0$ ,
- (e)  $q_0 = 19.0, q_1 = 14.0, q_2 = -4.1$ ,
- (f)  $q_0 = 14.0, q_1 = 19.0, q_2 = -4.1$ .

In Fig. 26, we see the obtained polynomiographs. From these polynomiographs, we can observe that although we used parameters which led to non-attractive patterns, their combinations in (20) allow us to obtain a variety of interesting patterns. Each of the obtained patterns has three visible braids, but they significantly differ in details.

Another way of defining  $\alpha_i$  for artistic applications is the use of linear interpolation between two given

values, i.e.

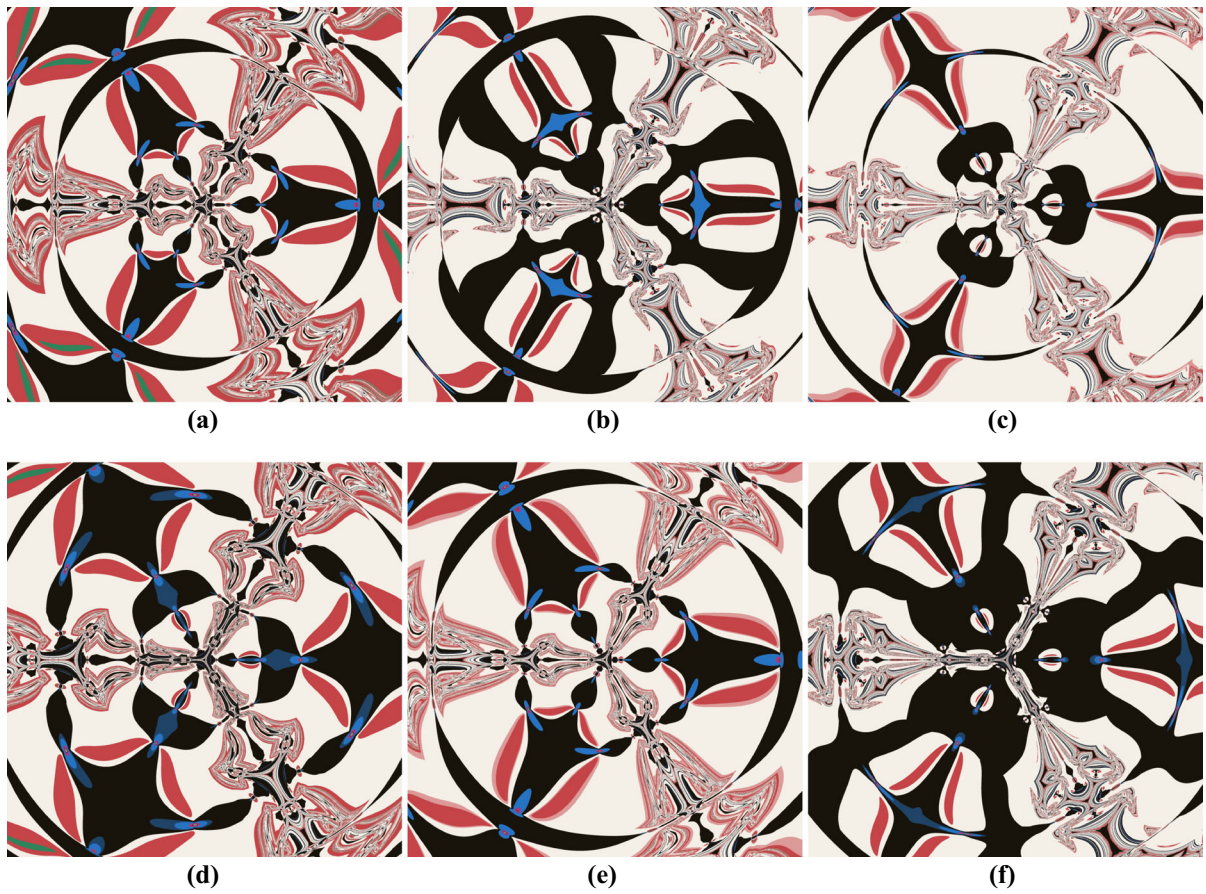
$$\alpha_i = q_0 \left( 1 - \frac{i}{M} \right) + q_1 \frac{i}{M}, \quad (21)$$

where  $q_0, q_1 \in \mathbb{R}$  and  $M$  is the maximum number of iterations.

Let us consider two values  $-17.0$  and  $30.0$  and the following parameters for the polynomiographs:  $p(z) = z^4 + z^2 - 1$ ,  $A = [-3, 3]^2$ ,  $M = 50$  and  $\varepsilon = 0.1$ . Similarly like in the previous example, let us start with the polynomiographs generated for the two considered values, see Fig. 27. These two polynomiographs present two completely different patterns. In Fig. 27a, we see a very simple and small pattern in the centre of the polynomiograph, whereas in Fig. 27b, the pattern occupies the whole area and it is rich in details. When we modify  $\alpha_i$  by using the linear combination of these two values: (a)  $q_0 = -17.0, q_1 = 30.0$ , (b)  $q_0 = 30.0, q_1 = -17.0$ , then we obtain the polynomiographs presented in Fig. 28. We see that both polynomiographs differ from the ones presented in Fig. 27. However, we can notice that some characteristic structures were inherited from the original patterns, i.e. in the pattern from Fig. 28a, we see the small pattern from Fig. 27a, and in the pattern from Fig. 28b, we see that the overall shape is similar to the pattern from Fig. 27b.

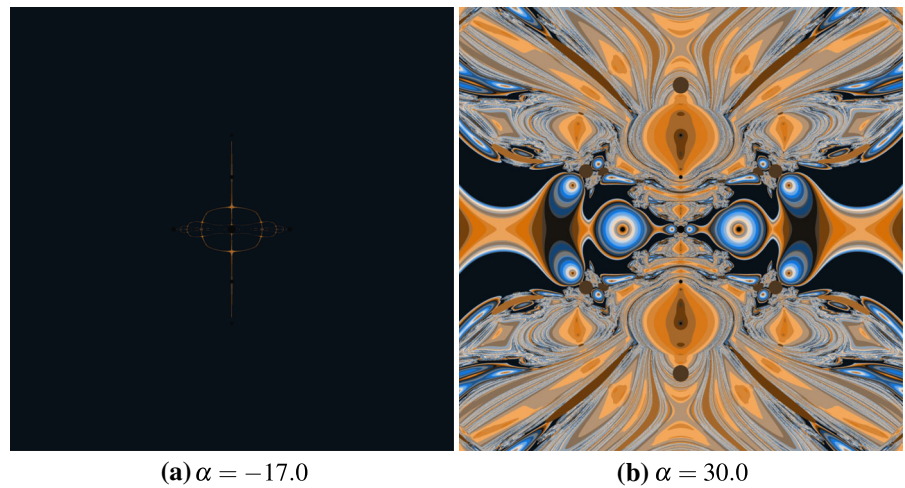
The linear interpolation applied in (21) can be used in many different ways to define  $\alpha_i$ . For instance, let us notice that when we generate a polynomiograph the  $\frac{i}{M}$  term in (21) belongs to  $[0, 1]$ . We can divide  $[0, 1]$  into non-overlapping intervals, and on each of them, we can use linear interpolation with different values of  $q_0$  and  $q_1$ . Basing on the parameters used in the previous





**Fig. 26** Dynamics polynomiographs for  $p(z) = z^3 - 1$  generated with the M-RNM, where  $\alpha_i$  is given by (20)

**Fig. 27** Dynamics polynomiographs for  $p(z) = z^4 + z^2 - 1$  generated with the M-RNM and various values of  $\alpha$

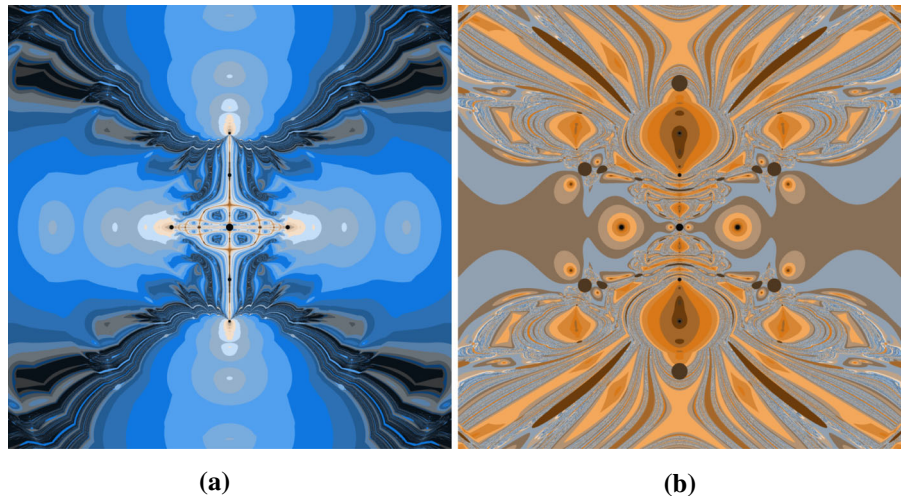


**(a)**  $\alpha = -17.0$

**(b)**  $\alpha = 30.0$



**Fig. 28** Dynamics polynomiographs for  $p(z) = z^4 + z^2 - 1$  generated with the M-RNM, where  $\alpha_i$  is given by (21)



example, we will present some polynomiographs that use this approach. Let us consider two sequences of  $\alpha_i$  defined in the following way:

$$\alpha_i = \begin{cases} -17.0 \left(1 - \frac{i}{M}\right) + 30.0 \frac{i}{M}, & \text{if } \frac{i}{M} < T, \\ 30.0 \left(1 - \frac{i}{M}\right) - 17.0 \frac{i}{M}, & \text{if } \frac{i}{M} \geq T, \end{cases} \quad (22)$$

$$\alpha_i = \begin{cases} 30.0 \left(1 - \frac{i}{M}\right) - 17.0 \frac{i}{M}, & \text{if } \frac{i}{M} < T, \\ -17.0 \left(1 - \frac{i}{M}\right) + 30.0 \frac{i}{M}, & \text{if } \frac{i}{M} \geq T, \end{cases} \quad (23)$$

where  $T \in [0, 1]$  is a threshold value that divides  $[0, 1]$  into two non-overlapping intervals. These two sequences differ only in the order of  $q_0$  and  $q_1$  values. Figures 29 and 30 present the polynomiographs obtained with these two sequences and varying values of  $T$ . From these polynomiographs, we can see that by changing the division of  $[0, 1]$  by the value of  $T$ , we are able to obtain a variety of interesting patterns that differ from the ones generated in Figs. 27 and 28.

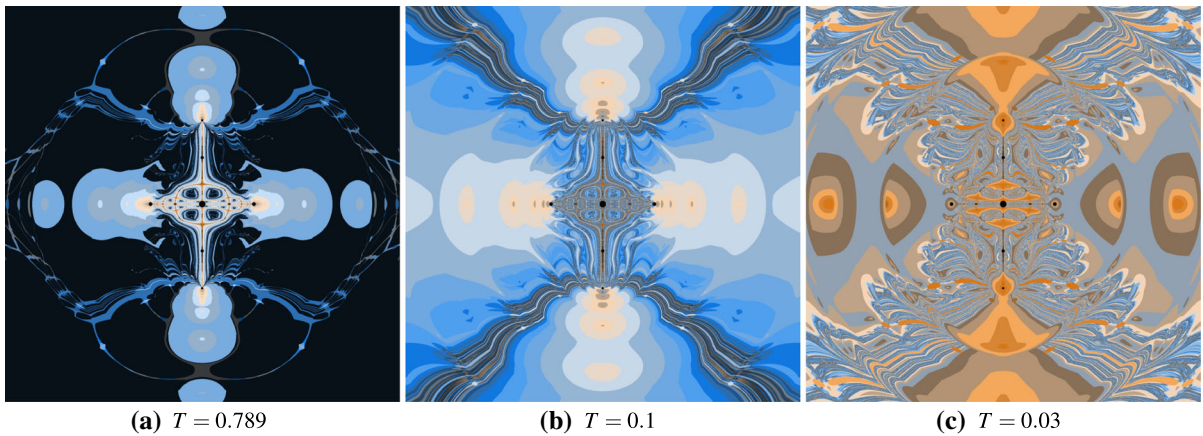
Recently, Bisheh-Niasar and Gdawiec [7] studied the use of periodic parameters in the  $S$ -iteration for the Bisheh-Niasar-Saadatmandi root finding method. For instance, they used simple trigonometric functions to define the parameters. By using this approach, they obtained some interesting patterns. In the case of the Mann iteration, considered in this paper, we can also use trigonometric functions in a variety of ways to define  $\alpha_i$ . In our example, we generated polynomiographs for the following parameters:  $p(z) = z^4 + 4$ ,  $A = [-3, 3]^2$ ,  $M = 200$  and  $\varepsilon = 0.001$ . The only parameters that vary among the polynomiographs are the colour map and Mann's iteration parameter  $\alpha_i$ . Figure 31 presents the polynomiographs for the following sequences of  $\alpha_i$ :

- (a)  $\alpha_i = 10 + 15 \cos(0.5i + 3.1415/2)$ ,
- (b)  $\alpha_i = 10 + 4 \sin(i) \tan(7.9i)$ ,
- (c)  $\alpha_i = 15 + 31 \sin(i) \cos(5i)$ ,
- (d)  $\alpha_i = 16 + 10 \sin(i) \tan(19i) - 6 \cos(i)$ ,
- (e)  $\alpha_i = 24 - \frac{4}{4.5}i + 15 \sin(i)$ ,
- (f)  $\alpha_i = 69 - \frac{34}{4.5}i + 25 \cos(2i)$ .

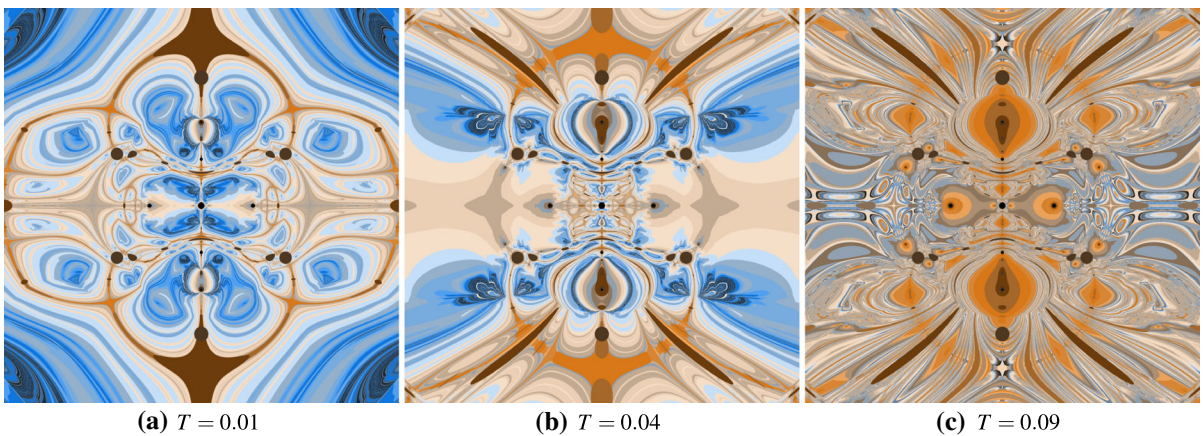
When we look at the polynomiographs in Fig. 31, we can see that the patterns differ in a great manner between each other, although they were generated from the same polynomial. They contain many fine details, which make these patterns very attractive from the artistic point of view.

In every presented example, we generated a single pattern, which can be used to create paintings, patterns on t-shirts, mugs, etc. However, when we look closely at these patterns, we can notice that with a proper selection of the area  $A$ , these patterns can serve as a template for a wallpaper or frieze symmetry patterns. Figure 32 presents examples of wallpaper symmetry patterns obtained from the templates generated by polynomiographs with the following parameters:

- (a)  $p(z) = z^4 + 4$ ,  $A = [-3, 3]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha_i = 24 - \frac{4}{4.5}i + 15 \sin(i)$ , (see Fig. 31e),
- (b)  $p(z) = z^4 - 1$ ,  $A = [-3, 3]^2$ ,  $M = 250$ ,  $\varepsilon = 0.001$  and  $\alpha_i = 37.75$ , (see Fig. 12k),
- (c)  $p(z) = z^3 - 1$ ,  $A = [-3, 3]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha = 17$ , (see Fig. 24c),
- (d)  $p(z) = z^6 - 1$ ,  $A = [-4.1, 4.1]^2$ ,  $M = 60$ ,  $\varepsilon = 0.1$  and  $\alpha_i = 25 + 30 \sin(i) \tan(40i)$  (see Fig. 33a),
- (e)  $p(z) = (4 + 4i)z^4 + 8iz^2 + 4$ ,  $A = [-3, 3]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha_i = 40 - \frac{13.5}{4.5}i$ , (see Fig. 33b),



**Fig. 29** Dynamics polynomiographs for  $p(z) = z^4 + z^2 - 1$  generated with the M-RNM, where  $\alpha_i$  is given by (22) and the  $T$  has various values



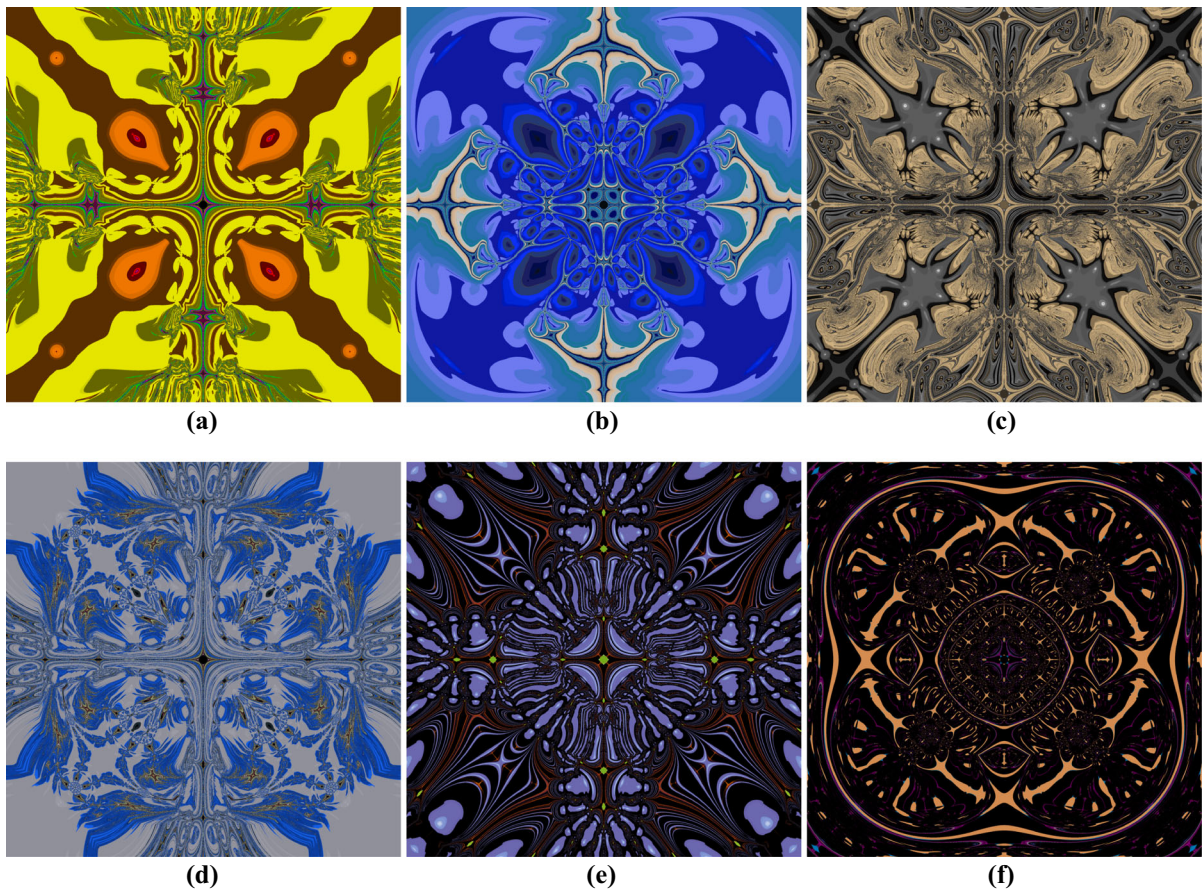
**Fig. 30** Dynamics polynomiographs for  $p(z) = z^4 + z^2 - 1$  generated with the M-RNM, where  $\alpha_i$  is given by (23) and the  $T$  has various values

- (f)  $p(z) = z^2 + 0.4375 + 1.5i$ ,  $A = [-3.7, 3.7]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha_i = 17.6 + 18.5 \sin(0.2i) \cos(10i)$ , (see Fig. 33c).

When we look at the templates in Figs. 31e and 12k, we see that they have a fourfold symmetry and four axes of symmetry (two diagonal, one horizontal and one vertical). Thus, using this kind of templates, we can create a wallpaper pattern of p4m symmetry (Fig. 32a, b) by repeating the template on a square lattice. If we place template from Fig. 24c on a square lattice, then we see that it has no translational symmetry nor glide symmetry. However, when between two adjacent cells in the square lattice we add a mirror reflection in the horizontal direction, then we obtain a pmm symmetry pattern (Fig. 32c). The template in Fig. 33a has a twofold

symmetry and two axes of symmetry (one horizontal and one vertical). These types of symmetry of the template will not give the p4m symmetry if we repeat it on a square lattice. Instead, we obtain a pmm symmetry pattern (Fig. 32d). A twofold symmetry is visible in the templates from Fig. 33b, c. These templates lack the horizontal and vertical mirror symmetries present in the template from Fig. 33a, so we cannot repeat them on a square lattice to obtain a pmm symmetry. However, we can use the mirror symmetries in horizontal and vertical directions between two adjacent cells in the square lattice to obtain a cmm symmetry patterns (Fig. 32e, f).





**Fig. 31** Dynamics polynomiographs for  $p(z) = z^4 + 4$  generated with the M-RNM, where  $\alpha_i$  is defined by using trigonometric functions

Next, Fig. 34 presents examples of frieze symmetry patterns obtained from the templates generated by the polynomiographs with the following parameters:

- (a)  $p(z) = z^3 - 1$ ,  $A = [-3.5, 3.5]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha_i$  given by (20) with  $m = 3$  and  $q_0 = -4.1$ ,  $q_1 = 19.0$ ,  $q_2 = 14.0$ , (see Fig. 26a),
- (b)  $p(z) = z^7 - 6z^5 - 31z^3 + 36z$ ,  $A = [-3.7, 3.7]^2$ ,  $M = 200$ ,  $\varepsilon = 0.001$  and  $\alpha_i = 30 + 10 \sin(i) \tan(8i)$ , (see Fig. 33d),
- (c)  $p(z) = z^4 + z^2 - 1$ ,  $A = [-3, 3]^2$ ,  $M = 50$ ,  $\varepsilon = 0.1$  and  $\alpha_i$  is given by (22) with  $T = 0.789$ , (see Fig. 29a).

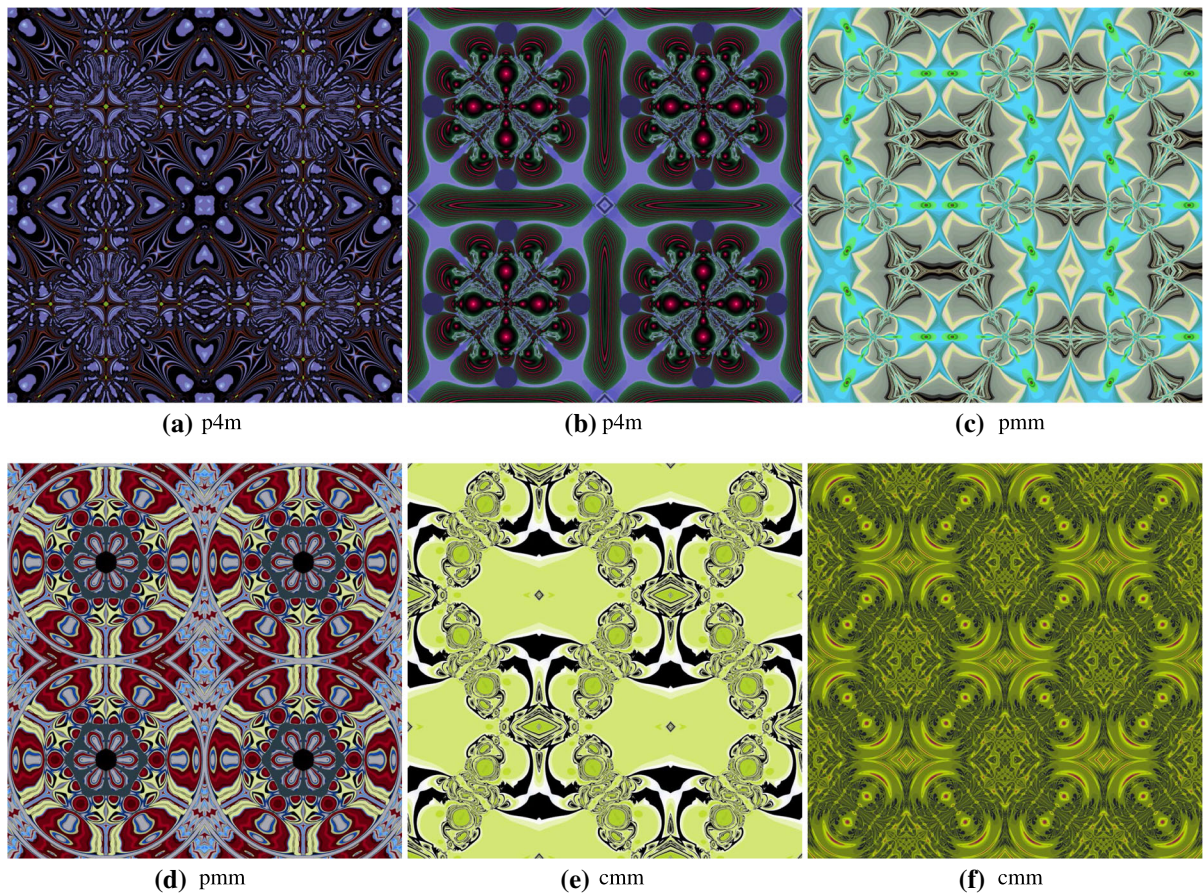
The template from Fig. 26a cannot be directly used to obtain a frieze symmetry pattern, because it lacks translational or glide symmetry after repeating it in a

horizontal direction. Nevertheless, if we rotate it by 90 degrees and repeat this template in a horizontal direction, then we obtain a pm1 symmetry pattern (Fig. 34a). When we look at the templates in Figs. 29a and 33d, we see that they have a twofold symmetry and two axes of symmetry (one horizontal and one vertical). Now, by repeating such templates in a horizontal direction, we obtain patterns of pmm symmetry (Fig. 34b, c).

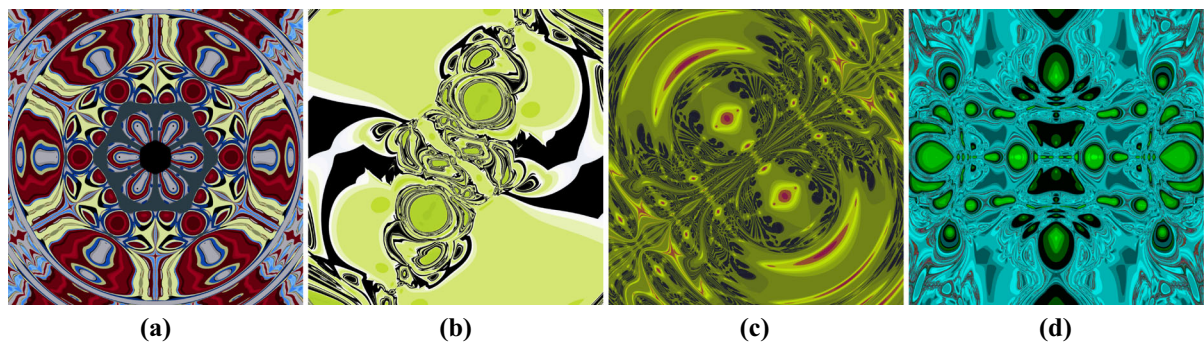
## 6 Conclusions

The RNM is a stable global root-finding process that needs a large number of iterations to achieve a root or a critical point of any polynomial. No chaos and no fractal structures are observable on the boundaries of basins when solving polynomial equations on the com-





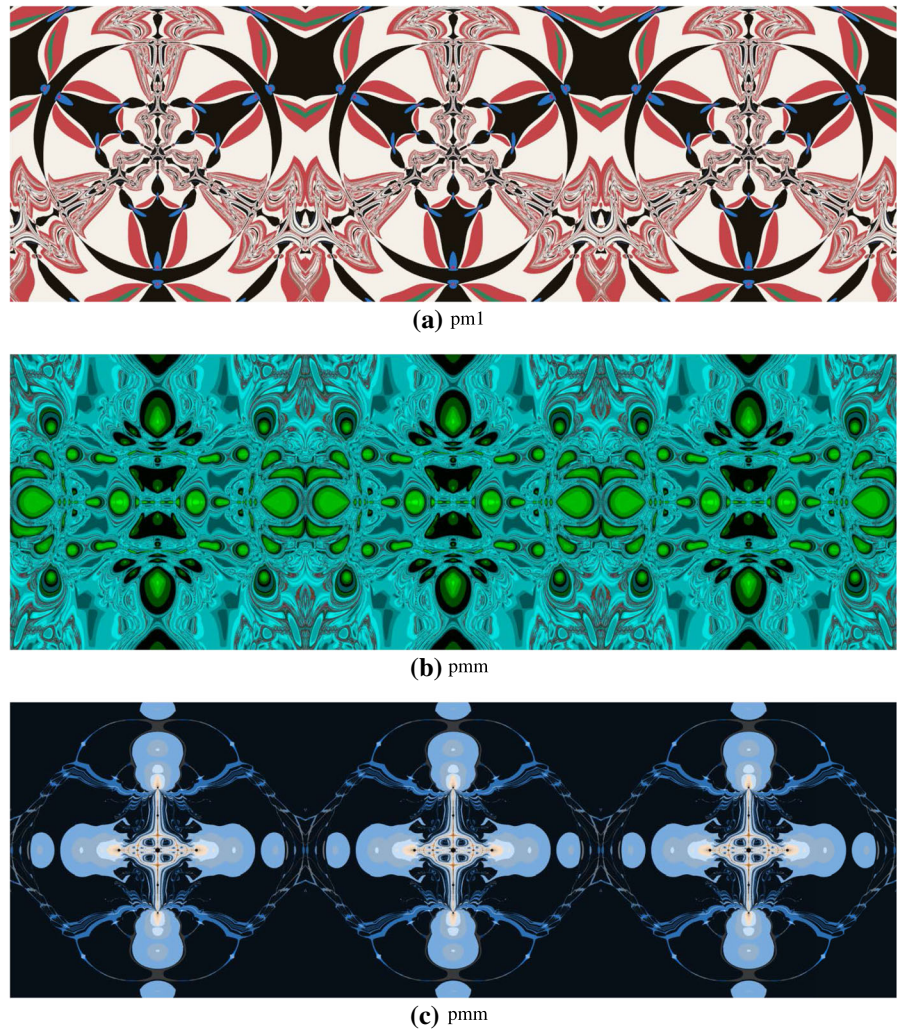
**Fig. 32** Wallpaper symmetry patterns obtained from the templates generated by the M-RNM



**Fig. 33** Polynomiographs generated by the M-RNM that can serve as templates for creating wallpaper or frieze symmetry patterns



**Fig. 34** Frieze symmetry patterns obtained from the templates generated by the M-RNM



plex plane, contrary to the classical Newton's method. In this paper, we modified the RNM by replacing the standard Picard iteration with the Mann iteration.

The proposed M-RNM root-finding algorithm significantly improves the RNM algorithm. The M-RNM is much faster than the RNM. Its speed of convergence depends on the parameter  $\alpha$  and the considered polynomial. The best value of  $\alpha$ , which assures good convergence and performance, can be determined experimentally for a given polynomial. The M-RNM algorithm does not lose good properties of the RNM such as global convergence, stability, and robustness even for relatively large deviations from the best value of  $\alpha$ . As the parameter  $\alpha$  increases and moves far away from the best value, the patterns become increasingly dynamic and eventually become more and more ran-

dom and chaotic. Also, for the proposed variable  $\alpha$  sequence, we can observe the good behaviour of the M-RNM, obtaining better results than the RNM. The appearance of the polynomiographs generated with the M-RNM can be easily modified by using various colour palettes. However, the artistic aspects of the images can be additionally achieved with the help of specially selected sequences of  $\alpha_i$  as in Sect. 5. Furthermore, these images have been successfully used as templates for wallpaper symmetry or frieze symmetry patterns giving nicely looking examples of applied art.

Encouraged by the improvement of the speed of the RNM algorithm by using the Mann iteration in place of Picard's iteration, we are thinking about checking whether the use of other types of iterations known from the literature and collected, e.g. in [13], can lead to

an even better improvement of the RNM algorithm. It also seems likely to find other intriguing patterns by using other types of iterations that may be considered aesthetic. Moreover, one can try to embed a proper template created with the M-RNM to the fundamental region of the spiral patterns introduced in [25]. These tasks will be the subject of our further research.

**Funding** This research received no external funding.

### Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

### References

- Agarwal, R.P., Karapinar, E., O'Regan, D., Roldán-Lopez-de Hierro, A.F.: *Fixed Points Theory in Metric Type Spaces*. Springer, Cham (2015)
- Akenine-Möller, T., Haines, E., Hoffman, N., Pesce, A., Iwanicki, M., Hillaire, S.: *Real-Time Rendering*, 4th edn. CRC Press, Boca Raton (2018)
- Akgül, A., Cordero, A., Torregrosa, J.R.: A fractional Newton method with  $2\alpha$ th-order of convergence and its stability. *Appl. Math. Lett.* **98**, 344–351 (2019)
- Ardelean, G., Balog, L.: A qualitative study of Agarwal et al. iteration procedure for fixed points approximation. *Creat. Math. Inform.* **25**(2), 135–139 (2016)
- Ardelean, G., Cosma, O., Balog, L.: A comparison of some fixed point iteration procedures by using the basins of attraction. *Carpath. J. Math.* **32**(3), 277–284 (2016)
- Berinde, V.: *Iterative Approximation of Fixed Points*. Springer, Berlin (2007)
- Bisheh-Niasar, M., Gdawiec, K.: Bisheh-Niasar-Saadatmandi root finding method via the  $S$ -iteration with periodic parameters and its polynomiography. *Math. Comput. Simul.* **160**, 1–12 (2019)
- Brambila-Paz, F., Torres-Hernandez, A.: Fractional Newton-Raphson method. [arxiv:1710.07634](https://arxiv.org/abs/1710.07634) (2017)
- Chung, K.W., Chan, H.S.Y.: Symmetrical patterns from dynamics. *Comput. Graph. Forum* **12**(1), 33–40 (1993)
- Cordero, A., Franceschi, J., Torregrosa, J.R., Zagati, A.C.: A convex combination approach for mean-based variants of Newton's method. *Symmetry* **11**(9), Article ID 1106 (2019)
- Epureanu, B.I., Greenside, H.S.: Fractal basins of attraction associated with a damped Newton's method. *SIAM Rev.* **40**(1), 102–109 (1998)
- Gdawiec, K.: Fractal patterns from the dynamics of combined polynomial root finding methods. *Nonlinear Dyn.* **90**(4), 2457–2479 (2017)
- Gdawiec, K., Kotarski, W.: Polynomiography for the polynomial infinity norm via Kalantari's formula and nonstandard iterations. *Appl. Math. Comput.* **307**, 17–30 (2017)
- Gdawiec, K., Kotarski, W., Lisowska, A.: Visual analysis of the Newton's method with fractional order derivatives. *Symmetry* **11**(9), Article ID 1143 (2019)
- Hubbard, J., Schleicher, D., Sutherland, S.: How to find all roots of complex polynomials by Newton's method. *Invent. Math.* **146**, 1–33 (2001)
- Jeong, M., Kim, G.O., Kim, S.-A.: Dynamics of Newton's method for solving some equations. *Comput. Graph.* **26**(2), 271–279 (2002)
- Kalantari, B.: *Polynomial Root-Finding and Polynomiography*. World Scientific, Singapore (2009)
- Kalantari, B.: A geometric modulus principle for polynomials. *Am. Math. Mon.* **118**(10), 931–935 (2011)
- Kalantari, B.: A globally convergent Newton method for polynomials. [arxiv:2003.00372](https://arxiv.org/abs/2003.00372) (2020)
- Lu, J., Ye, Z., Zou, Y.: Automatic generation of colorful patterns with wallpaper symmetries from dynamics. *Vis. Comput.* **23**(6), 445–449 (2007)
- Mann, W.R.: Mean value methods in iteration. *Proc. Am. Math. Soc.* **4**(3), 506–510 (1953)
- May, R.M.: Simple mathematical models with very complicated dynamics. *Nature* **261**, 459–467 (1976)
- Murray, W.: Newton-type methods. In: Cochran, J.J., Cox, L.A., Keskinocak, P., Kharoufeh, J.P., Smith, J.C. (eds.) *Wiley Encyclopedia of Operations Research and Management Science*. Wiley, Hoboken (2011)
- Ortega, J.M., Rheinboldt, W.C.: *Iterative Solutions of Nonlinear Equations in Several Variables*. SIAM, Philadelphia (2000)
- Ouyang, P., Tang, X., Chung, K., Yu, T.: Spiral patterns of color symmetry from dynamics. *Nonlinear Dyn.* **94**(1), 261–272 (2018)
- Petković, I., Rančić, L.Z.: Computational geometry as a tool for studying root-finding methods. *Filomat* **33**(4), 1019–1027 (2019)
- Picard, E.: Mémoire sur la théorie des équations aux dérivées partielles et la méthode des approximations successives. *J. Math. Pures Appl.* **6**(4), 145–210 (1890)
- Polyak, B.T.: Newton's method and its use in optimization. *Eur. J. Oper. Res.* **181**(3), 1086–1096 (2007)
- Rhyné, T.-M.: *Applying Color Theory to Digital Media and Visualization*. CRC Press, Boca Raton (2017)
- Schleicher, D., Stoll, R.: Newton's method in practice: finding all roots of polynomials of degree one million efficiently. *Theor. Comput. Sci.* **681**, 146–166 (2017)
- Stanimirović, P.S., Ivanov, B., Ma, H., Mosić, D.: A survey of gradient methods for solving nonlinear optimization. *Electron. Res. Arch.* **28**(4), 1573–1624 (2020)

32. Stone, T.L., Adams, S., Morioka, N.: Color Design Workbook: A Real-World Guide to Using Color in Graphic Design. Rockport Publishers, Inc., Beverly (2006)
33. Wang, X., Tao, Y.: A new Newton method with memory for solving nonlinear equations. *Mathematics* **8**(1), 108 (2020)
34. Wang, X.-Y., Li, Y.-K., Sun, Y.-Y., Song, J.-M., Ge, F.-D.: Julia sets of Newton's method for a class of complex-exponential function  $F(z) = P(z)e^{Q(z)}$ . *Nonlinear Dyn.* **62**(4), 955–966 (2010)
35. Yau, L., Ben-Israel, A.: The Newton and Halley methods for complex roots. *Am. Math. Mon.* **105**(9), 806–818 (1998)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.